W271 Section 1: Lab3 Report

Shih Yu Chang, Nick Stamatakis, Tony Panza

December 8, 2016

1 Introduction

Many students who begin univeristy / college do not complete. There are many potential reasons for this: students may lack proper core education, schools may not provide adequate support or students may run into financial obstacles.

Presented below is a study to assess the impact financial support / college cost has on completion rates. We theorize that financial stress is a cause of dropping out of college. Therefore, we have two hypotheses:

a. Higher grant support increases completion rate
b. Lower total cost for education increases completion rate

There are two primary data sources for this analysis, IPEDS [5] and the College Scorecard [2]. Both are panel sets representing aggregate statistics for over 8000 colleges and universities in the United States. Standard OLS regression is used initially, and where the data allows, First Differences estimation is used to reduce OLS baises.

The final conclusion is that increasing financial support on a college level by either increasing grants or effectively lowering the cost has no impact on institutional completion rates, either with broad student populations or specifially low income students.

2 Lower Total Cost

Broadly, we are testing if a lower total cost for education increases completion rate. Specifically, we are studying the correlation between the cost of attendace vs. institutional 4 year completion rates, controlling for other factors of college completion (admission rate, average SAT score, acceptance rate). This will be tested against the full student population as well as against low income students alone.

Because there is additional available data, we will use a first differences approach to handle the omitted variaible bias and possible heterogeneity arrising from the use of panel data.

2.1 Data Source

The primary source of data is the College Scorecard [2]. The College Scorecard consolidates and metrics from a wide variety of government sources pertaining to numerous educational institutions across the country.

As an aggregation from multiple agencies, the College Scorecard is fairly incomplete, with much missing data, and missing variables in certain years.

An initial analysis of the variables we were interested in revealed that the widest possible range we could use with the least missing data was 2009 to 2012. The below table shows the data being drawn from the 2009 and 2012 datasets.

```
### Data Dictionary
# UNITID: Unit ID for institution
# INSTNM: Institution name

######## PRIMARY INDEPENDENT VARIABLES

# The average annual total cost of attendance (CostT4_A, CostT4_P),
# including tuition and fees, books and supplies, and living expenses, minus the average
# grant/scholarship aid, by detailed income category. It is calculated for all full-time,
# first-time, degree/certificate-seeking undergraduates who receive Title IV aid.

# NPT41_PUB: Average net price for $0-$30,000 family income (public institutions)
# NPT41_PRIV: Average net price for $0-$30,000 family income (private for-profit and nonprofit institut
# NPT41_PROG: Average net price for $0-$30,000 family income (program-year institutions)
# NPT41_OTHER: Average net price for $0-$30,000 family income (other academic calendar institutions)

# "low-income" defined as: less than $30,000 in nominal family income

# NPT4_PUB: Average net price (all incomes) (public institutions)
# NPT41_PRIV: Average net price (all incomes) (private for-profit and nonprofit institutions)
# NPT41_PROG: Average net price (all incomes) (program-year institutions)
# NPT41_OTHER: Average net price (all incomes) (other academic calendar institutions)

######## PRIMARY DEPENDENT VARIABLES

# LO_INC_COMP_ORIG_YR2_RT: Percent of low-income students who completed within 2 years at original inst
# LO_INC_COMP_ORIG_YR3_RT: Percent of low-income students who completed within 3 years at original inst
# LO_INC_COMP_ORIG_YR4_RT: Percent of low-income students who completed within 4 years at original inst
# LO_INC_COMP_ORIG_YR6_RT: Percent of low-income students who completed within 6 years at original inst
# LO_INC_COMP_ORIG_YR8_RT: Percent of low-income students who completed within 8 years at original inst

# LO_INC_COMP_4YR_TRANS_YR2_RT: Percent of low-income students who transferred to a 4-year institution
# LO_INC_COMP_4YR_TRANS_YR3_RT: Percent of low-income students who transferred to a 4-year institution
# LO_INC_COMP_4YR_TRANS_YR4_RT: Percent of low-income students who transferred to a 4-year institution
# LO_INC_COMP_4YR_TRANS_YR6_RT: Percent of low-income students who transferred to a 4-year institution
# LO_INC_COMP_4YR_TRANS_YR8_RT: Percent of low-income students who transferred to a 4-year institution

# LO_INC_COMP_2YR_TRANS_YR2_RT: Percent of low-income students who transferred to a 2-year institution
# LO_INC_COMP_2YR_TRANS_YR3_RT: Percent of low-income students who transferred to a 2-year institution
# LO_INC_COMP_2YR_TRANS_YR4_RT: Percent of low-income students who transferred to a 2-year institution
# LO_INC_COMP_2YR_TRANS_YR6_RT: Percent of low-income students who transferred to a 2-year institution
# LO_INC_COMP_2YR_TRANS_YR8_RT: Percent of low-income students who transferred to a 2-year institution

# COMP_ORIG_YR4_RT - Four year completion rates (full population)
# COMP_ORIG_YR6_RT - Six year completion rates (full population)

######## CONTROL VARIABLES

# CCUGPROF - Type of institution (3 = 4 year bachelors)
# SAT_AVG - Average SAT Score
# ADM_RATE - Total admission rate
# INEXPFTE - Amount spent on instruction
# LO_INC_YR, MD_INC_YR, HI_INC_YR - Number of students in graduation cohorts at different incomes (to p
```

```
###
```

2.2 Data Importation and Cleaning

To produce a final analysis_frame, substantial transformations and cleaning is required.

```
MERGED2012_PP <- read.csv("~/R/Data/MERGED2012_PP.csv")
MERGED2009_PP <- read.csv("~/R/Data/MERGED2009_PP.csv")

# sometimes colname for UNITID is read as ?..UNITID (i with two dots on top)
# other times that column is named ?..UNITID
# rename it to just UNITID, regardless of what the leading characters are
names(MERGED2012_PP) <- sub("^.*UNITID$", "UNITID", names(MERGED2012_PP))
names(MERGED2009_PP) <- sub("^.*UNITID$", "UNITID", names(MERGED2009_PP))


colsearch_09 <- colnames(MERGED2009_PP)
colsearch_12 <- colnames(MERGED2012_PP)

Merged_09 <- MERGED2009_PP[,c(colsearch_09[grep("LO_INC_COMP", colsearch_09)],
                              colsearch_09[grep("NPT41", colsearch_09)],
                              colsearch_09[grep("NPT4_", colsearch_09)],
                              colsearch_09[grep("LO_INC_YR", colsearch_09)],
                              colsearch_09[grep("MD_INC_YR", colsearch_09)],
                              colsearch_09[grep("HI_INC_YR", colsearch_09)],
                      "UNITID","INSTNM","OPEID","PREDDEG","CCUGPROF","SAT_AVG","ADM_RATE","INEXPFTE","C

Merged_12 <- MERGED2012_PP[,c(colsearch_12[grep("LO_INC_COMP", colsearch_12)],
                              colsearch_12[grep("NPT41", colsearch_12)],
                              colsearch_12[grep("NPT4_", colsearch_12)],
                              colsearch_12[grep("LO_INC_YR", colsearch_12)],
                              colsearch_12[grep("MD_INC_YR", colsearch_12)],
                              colsearch_09[grep("HI_INC_YR", colsearch_12)],


main_frame <- merge(Merged_09, Merged_12, by.x = "UNITID", by.y = "UNITID")


# Some values throughout many columns in main_frame are missing
# In some cases, these are coded as "NULL".
# In other cases, these are coded as "PrivacySuppressed".
# Recoded both of these to NA
main_frame[main_frame == "NULL"] <- NA
main_frame[main_frame == "PrivacySuppressed"] <- NA

# convert all NPT4*.x cols to numemric
main_frame$NPT41_PUB.x <- as.numeric(as.character(main_frame[,c("NPT41_PUB.x")]))
main_frame$NPT41_PRIV.x <- as.numeric(as.character(main_frame[,c("NPT41_PRIV.x")]))
main_frame$NPT41_PROG.x <- as.numeric(as.character(main_frame[,c("NPT41_PROG.x")]))
main_frame$NPT41_OTHER.x <- as.numeric(as.character(main_frame[,c("NPT41_OTHER.x")]))

main_frame$NPT4_PUB.x <- as.numeric(as.character(main_frame[,c("NPT4_PUB.x")]))
main_frame$NPT4_PRIV.x <- as.numeric(as.character(main_frame[,c("NPT4_PRIV.x")]))
```

```
main_frame$NPT4_PROG.x <- as.numeric(as.character(main_frame[,c("NPT4_PROG.x")]))
main_frame$NPT4_OTHER.x <- as.numeric(as.character(main_frame[,c("NPT4_OTHER.x")]))

# now that all of the NPT4_*.x cols are numeric, merge them into one,
# since they are mutually exclusive (only one can be non-NA)
main_frame$NPT41.x<-rowSums(main_frame[, c("NPT41_PUB.x", "NPT41_PRIV.x", "NPT41_PROG.x", "NPT41_OTHER.
main_frame$NPT4.x<-rowSums(main_frame[, c("NPT4_PUB.x", "NPT4_PRIV.x", "NPT4_PROG.x", "NPT4_OTHER.x")],

# if all of the NPT41_*.x cols were NA in a row, then NPT41.x is 0. convert these rows back to NA
main_frame$NPT41.x[main_frame$NPT41.x == 0] <- NA
main_frame$NPT4.x[main_frame$NPT41.x == 0] <- NA

# convert all NPT4_*.y cols to numemric
main_frame$NPT41_PUB.y <- as.numeric(as.character(main_frame[,c("NPT41_PUB.y")]))
main_frame$NPT41_PRIV.y <- as.numeric(as.character(main_frame[,c("NPT41_PRIV.y")]))
main_frame$NPT41_PROG.y <- as.numeric(as.character(main_frame[,c("NPT41_PROG.y")]))
main_frame$NPT41_OTHER.y <- as.numeric(as.character(main_frame[,c("NPT41_OTHER.y")]))

main_frame$NPT4_PUB.y <- as.numeric(as.character(main_frame[,c("NPT4_PUB.y")]))
main_frame$NPT4_PRIV.y <- as.numeric(as.character(main_frame[,c("NPT4_PRIV.y")]))
main_frame$NPT4_PROG.y <- as.numeric(as.character(main_frame[,c("NPT4_PROG.y")]))
main_frame$NPT4_OTHER.y <- as.numeric(as.character(main_frame[,c("NPT4_OTHER.y")]))


# now that all of the NPT41_*.y cols are numeric, merge them into one,
# since they are mutually exclusive (only one can be non-NA)
main_frame$NPT41.y<-rowSums(main_frame[, c("NPT41_PUB.y", "NPT41_PRIV.y", "NPT41_PROG.y", "NPT41_OTHER.y
main_frame$NPT4.y<-rowSums(main_frame[, c("NPT4_PUB.y", "NPT4_PRIV.y", "NPT4_PROG.y", "NPT4_OTHER.y")],


# if all of the NPT41_*.y cols were NA in a row, then NPT41.y is 0. convert these rows back to NA
main_frame$NPT41.y[main_frame$NPT41.y == 0] <- NA
main_frame$NPT4.y[main_frame$NPT4.y == 0] <- NA

# Convert the Completion rates to numbers
main_frame$LO_INC_COMP_ORIG_YR2_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_ORIG_YR2_RT.
main_frame$LO_INC_COMP_4YR_TRANS_YR2_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_4YR_TRA
main_frame$LO_INC_COMP_2YR_TRANS_YR2_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_2YR_TRA
main_frame$LO_INC_COMP_ORIG_YR3_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_ORIG_YR3_RT.
main_frame$LO_INC_COMP_4YR_TRANS_YR3_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_4YR_TRA
main_frame$LO_INC_COMP_2YR_TRANS_YR3_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_2YR_TRA
main_frame$LO_INC_COMP_ORIG_YR4_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_ORIG_YR4_RT.
main_frame$LO_INC_COMP_4YR_TRANS_YR4_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_4YR_TRA
main_frame$LO_INC_COMP_2YR_TRANS_YR4_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_2YR_TRA
main_frame$LO_INC_COMP_ORIG_YR6_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_ORIG_YR6_RT.
main_frame$LO_INC_COMP_4YR_TRANS_YR6_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_4YR_TRA
main_frame$LO_INC_COMP_2YR_TRANS_YR6_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_2YR_TRA
main_frame$LO_INC_COMP_ORIG_YR8_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_ORIG_YR8_RT.
main_frame$LO_INC_COMP_4YR_TRANS_YR8_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_4YR_TRA
main_frame$LO_INC_COMP_2YR_TRANS_YR8_RT.x <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_2YR_TRA

main_frame$LO_INC_COMP_ORIG_YR2_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_ORIG_YR2_RT.
main_frame$LO_INC_COMP_4YR_TRANS_YR2_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_4YR_TRA
```

```
main_frame$LO_INC_COMP_2YR_TRANS_YR2_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_2YR_TRAN
main_frame$LO_INC_COMP_ORIG_YR3_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_ORIG_YR3_RT.y
main_frame$LO_INC_COMP_4YR_TRANS_YR3_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_4YR_TRAN
main_frame$LO_INC_COMP_2YR_TRANS_YR3_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_2YR_TRAN
main_frame$LO_INC_COMP_ORIG_YR4_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_ORIG_YR4_RT.y
main_frame$LO_INC_COMP_4YR_TRANS_YR4_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_4YR_TRAN
main_frame$LO_INC_COMP_2YR_TRANS_YR4_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_2YR_TRAN
main_frame$LO_INC_COMP_ORIG_YR6_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_ORIG_YR6_RT.y
main_frame$LO_INC_COMP_4YR_TRANS_YR6_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_4YR_TRAN
main_frame$LO_INC_COMP_2YR_TRANS_YR6_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_2YR_TRAN
main_frame$LO_INC_COMP_ORIG_YR8_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_ORIG_YR8_RT.y
main_frame$LO_INC_COMP_4YR_TRANS_YR8_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_4YR_TRAN
main_frame$LO_INC_COMP_2YR_TRANS_YR8_RT.y <- as.numeric(as.character(main_frame[,c("LO_INC_COMP_2YR_TRAN



# convert cohort values
main_frame$LO_INC_YR2_N.x <- as.numeric(as.character(main_frame[,c("LO_INC_YR2_N.x")]))
main_frame$LO_INC_YR3_N.x <- as.numeric(as.character(main_frame[,c("LO_INC_YR3_N.x")]))
main_frame$LO_INC_YR4_N.x <- as.numeric(as.character(main_frame[,c("LO_INC_YR4_N.x")]))
main_frame$LO_INC_YR6_N.x <- as.numeric(as.character(main_frame[,c("LO_INC_YR6_N.x")]))
main_frame$LO_INC_YR8_N.x <- as.numeric(as.character(main_frame[,c("LO_INC_YR8_N.x")]))

main_frame$MD_INC_YR2_N.x <- as.numeric(as.character(main_frame[,c("MD_INC_YR2_N.x")]))
main_frame$MD_INC_YR3_N.x <- as.numeric(as.character(main_frame[,c("MD_INC_YR3_N.x")]))
main_frame$MD_INC_YR4_N.x <- as.numeric(as.character(main_frame[,c("MD_INC_YR4_N.x")]))
main_frame$MD_INC_YR6_N.x <- as.numeric(as.character(main_frame[,c("MD_INC_YR6_N.x")]))
main_frame$MD_INC_YR8_N.x <- as.numeric(as.character(main_frame[,c("MD_INC_YR8_N.x")]))

main_frame$HI_INC_YR2_N.x <- as.numeric(as.character(main_frame[,c("HI_INC_YR2_N.x")]))
main_frame$HI_INC_YR3_N.x <- as.numeric(as.character(main_frame[,c("HI_INC_YR3_N.x")]))
main_frame$HI_INC_YR4_N.x <- as.numeric(as.character(main_frame[,c("HI_INC_YR4_N.x")]))
main_frame$HI_INC_YR6_N.x <- as.numeric(as.character(main_frame[,c("HI_INC_YR6_N.x")]))
main_frame$HI_INC_YR8_N.x <- as.numeric(as.character(main_frame[,c("HI_INC_YR8_N.x")]))

main_frame$LO_INC_YR2_N.y <- as.numeric(as.character(main_frame[,c("LO_INC_YR2_N.y")]))
main_frame$LO_INC_YR3_N.y <- as.numeric(as.character(main_frame[,c("LO_INC_YR3_N.y")]))
main_frame$LO_INC_YR4_N.y <- as.numeric(as.character(main_frame[,c("LO_INC_YR4_N.y")]))
main_frame$LO_INC_YR6_N.y <- as.numeric(as.character(main_frame[,c("LO_INC_YR6_N.y")]))
main_frame$LO_INC_YR8_N.y <- as.numeric(as.character(main_frame[,c("LO_INC_YR8_N.y")]))

main_frame$MD_INC_YR2_N.y <- as.numeric(as.character(main_frame[,c("MD_INC_YR2_N.y")]))
main_frame$MD_INC_YR3_N.y <- as.numeric(as.character(main_frame[,c("MD_INC_YR3_N.y")]))
main_frame$MD_INC_YR4_N.y <- as.numeric(as.character(main_frame[,c("MD_INC_YR4_N.y")]))
main_frame$MD_INC_YR6_N.y <- as.numeric(as.character(main_frame[,c("MD_INC_YR6_N.y")]))
main_frame$MD_INC_YR8_N.y <- as.numeric(as.character(main_frame[,c("MD_INC_YR8_N.y")]))

main_frame$HI_INC_YR2_N.y <- as.numeric(as.character(main_frame[,c("HI_INC_YR2_N.y")]))
main_frame$HI_INC_YR3_N.y <- as.numeric(as.character(main_frame[,c("HI_INC_YR3_N.y")]))
main_frame$HI_INC_YR4_N.y <- as.numeric(as.character(main_frame[,c("HI_INC_YR4_N.y")]))
main_frame$HI_INC_YR6_N.y <- as.numeric(as.character(main_frame[,c("HI_INC_YR6_N.y")]))
main_frame$HI_INC_YR8_N.y <- as.numeric(as.character(main_frame[,c("HI_INC_YR8_N.y")]))
```

```
main_frame$SAT_AVG.y <- as.numeric(as.character(main_frame[,c("SAT_AVG.y")]))
main_frame$SAT_AVG.x <- as.numeric(as.character(main_frame[,c("SAT_AVG.x")]))

main_frame$ADM_RATE.x <- as.numeric(as.character(main_frame[,c("ADM_RATE.x")]))
main_frame$ADM_RATE.y <- as.numeric(as.character(main_frame[,c("ADM_RATE.y")]))

main_frame$INEXPFTE.x <- as.numeric(as.character(main_frame[,c("INEXPFTE.x")]))
main_frame$INEXPFTE.y <- as.numeric(as.character(main_frame[,c("INEXPFTE.y")]))

main_frame$COMP_ORIG_YR4_RT.x <- as.numeric(as.character(main_frame[,c("COMP_ORIG_YR4_RT.x")]))
main_frame$COMP_ORIG_YR4_RT.y <- as.numeric(as.character(main_frame[,c("COMP_ORIG_YR4_RT.y")]))

### Calculate Completion Rates
main_frame$yr4.x <-
(
ifelse(is.na(main_frame$LO_INC_COMP_ORIG_YR4_RT.x),0,main_frame$LO_INC_COMP_ORIG_YR4_RT.x)
+
ifelse(is.na(main_frame$LO_INC_COMP_4YR_TRANS_YR4_RT.x),0,main_frame$LO_INC_COMP_4YR_TRANS_YR4_RT.x)
+
ifelse(is.na(main_frame$LO_INC_COMP_2YR_TRANS_YR4_RT.x),0,main_frame$LO_INC_COMP_2YR_TRANS_YR4_RT.x)
)

main_frame$yr4.y <-
(
ifelse(is.na(main_frame$LO_INC_COMP_ORIG_YR4_RT.y),0,main_frame$LO_INC_COMP_ORIG_YR4_RT.y)
+
ifelse(is.na(main_frame$LO_INC_COMP_4YR_TRANS_YR4_RT.y),0,main_frame$LO_INC_COMP_4YR_TRANS_YR4_RT.y)
+
ifelse(is.na(main_frame$LO_INC_COMP_2YR_TRANS_YR4_RT.y),0,main_frame$LO_INC_COMP_2YR_TRANS_YR4_RT.y)
)

main_frame$yr6.x <-
(
ifelse(is.na(main_frame$LO_INC_COMP_ORIG_YR6_RT.x),0,main_frame$LO_INC_COMP_ORIG_YR6_RT.x)
+
ifelse(is.na(main_frame$LO_INC_COMP_4YR_TRANS_YR6_RT.x),0,main_frame$LO_INC_COMP_4YR_TRANS_YR6_RT.x)
+
ifelse(is.na(main_frame$LO_INC_COMP_2YR_TRANS_YR6_RT.x),0,main_frame$LO_INC_COMP_2YR_TRANS_YR6_RT.x)
)

main_frame$yr6.y <-
(
ifelse(is.na(main_frame$LO_INC_COMP_ORIG_YR6_RT.y),0,main_frame$LO_INC_COMP_ORIG_YR6_RT.y)
+
ifelse(is.na(main_frame$LO_INC_COMP_4YR_TRANS_YR6_RT.y),0,main_frame$LO_INC_COMP_4YR_TRANS_YR6_RT.y)
+
ifelse(is.na(main_frame$LO_INC_COMP_2YR_TRANS_YR6_RT.y),0,main_frame$LO_INC_COMP_2YR_TRANS_YR6_RT.y)
)

main_frame$yr8.x <-
(
ifelse(is.na(main_frame$LO_INC_COMP_ORIG_YR8_RT.x),0,main_frame$LO_INC_COMP_ORIG_YR8_RT.x)
+
```

```
ifelse(is.na(main_frame$LO_INC_COMP_4YR_TRANS_YR8_RT.x),0,main_frame$LO_INC_COMP_4YR_TRANS_YR8_RT.x)
+
ifelse(is.na(main_frame$LO_INC_COMP_2YR_TRANS_YR8_RT.x),0,main_frame$LO_INC_COMP_2YR_TRANS_YR8_RT.x)
)

main_frame$yr8.y <-
(
ifelse(is.na(main_frame$LO_INC_COMP_ORIG_YR8_RT.y),0,main_frame$LO_INC_COMP_ORIG_YR8_RT.y)
+
ifelse(is.na(main_frame$LO_INC_COMP_4YR_TRANS_YR8_RT.y),0,main_frame$LO_INC_COMP_4YR_TRANS_YR8_RT.y)
+
ifelse(is.na(main_frame$LO_INC_COMP_2YR_TRANS_YR8_RT.y),0,main_frame$LO_INC_COMP_2YR_TRANS_YR8_RT.y)
)

## Agregate Cohort statistics
main_frame$cohortsize_8.x <-
  ifelse(is.na(main_frame$LO_INC_YR8_N.x),0,main_frame$LO_INC_YR8_N.x) +
  ifelse(is.na(main_frame$MD_INC_YR8_N.x),0,main_frame$MD_INC_YR8_N.x) +
  ifelse(is.na(main_frame$HI_INC_YR8_N.x),0,main_frame$HI_INC_YR8_N.x)
main_frame$cohortsize_8.y <-
  ifelse(is.na(main_frame$LO_INC_YR8_N.y),0,main_frame$LO_INC_YR8_N.y) +
  ifelse(is.na(main_frame$MD_INC_YR8_N.y),0,main_frame$MD_INC_YR8_N.y) +
  ifelse(is.na(main_frame$HI_INC_YR8_N.y),0,main_frame$HI_INC_YR8_N.y)

main_frame$cohortsize_6.x <-
  ifelse(is.na(main_frame$LO_INC_YR6_N.x),0,main_frame$LO_INC_YR6_N.x) +
  ifelse(is.na(main_frame$MD_INC_YR6_N.x),0,main_frame$MD_INC_YR6_N.x) +
  ifelse(is.na(main_frame$HI_INC_YR6_N.x),0,main_frame$HI_INC_YR6_N.x)
main_frame$cohortsize_6.y <-
  ifelse(is.na(main_frame$LO_INC_YR6_N.y),0,main_frame$LO_INC_YR6_N.y) +
  ifelse(is.na(main_frame$MD_INC_YR6_N.y),0,main_frame$MD_INC_YR6_N.y) +
  ifelse(is.na(main_frame$HI_INC_YR6_N.y),0,main_frame$HI_INC_YR6_N.y)

main_frame$cohortsize_4.x <-
  ifelse(is.na(main_frame$LO_INC_YR4_N.x),0,main_frame$LO_INC_YR4_N.x) +
  ifelse(is.na(main_frame$MD_INC_YR4_N.x),0,main_frame$MD_INC_YR4_N.x) +
  ifelse(is.na(main_frame$HI_INC_YR4_N.x),0,main_frame$HI_INC_YR4_N.x)
main_frame$cohortsize_4.y <-
  ifelse(is.na(main_frame$LO_INC_YR4_N.y),0,main_frame$LO_INC_YR4_N.y) +
  ifelse(is.na(main_frame$MD_INC_YR4_N.y),0,main_frame$MD_INC_YR4_N.y) +
  ifelse(is.na(main_frame$HI_INC_YR4_N.y),0,main_frame$HI_INC_YR4_N.y)

main_frame$percentlow_8.x <- main_frame$LO_INC_YR8_N.x / main_frame$cohortsize_8.x
main_frame$percentlow_8.y <- main_frame$LO_INC_YR8_N.y / main_frame$cohortsize_8.y

main_frame$percentlow_6.x <- main_frame$LO_INC_YR6_N.x / main_frame$cohortsize_6.x
main_frame$percentlow_6.y <- main_frame$LO_INC_YR6_N.y / main_frame$cohortsize_6.y

main_frame$percentlow_4.x <- main_frame$LO_INC_YR4_N.x / main_frame$cohortsize_4.x
main_frame$percentlow_4.y <- main_frame$LO_INC_YR4_N.y / main_frame$cohortsize_4.y


## Identify records with full cohort sizes
```

```
main_frame$hi_size_all_8 <-
  ifelse(ifelse(is.na(main_frame$HI_INC_YR8_N.y),0,main_frame$HI_INC_YR8_N.y) == 0, 0,
  ifelse(ifelse(is.na(main_frame$HI_INC_YR8_N.x),0,main_frame$HI_INC_YR8_N.x) == 0, 0, 1))
main_frame$med_size_all_8 <-
  ifelse(ifelse(is.na(main_frame$MD_INC_YR8_N.y),0,main_frame$MD_INC_YR8_N.y) == 0, 0,
  ifelse(ifelse(is.na(main_frame$MD_INC_YR8_N.x),0,main_frame$MD_INC_YR8_N.x) == 0, 0, 1))
main_frame$low_size_all_8 <-
  ifelse(ifelse(is.na(main_frame$LO_INC_YR8_N.y),0,main_frame$LO_INC_YR8_N.y) == 0, 0,
  ifelse(ifelse(is.na(main_frame$LO_INC_YR8_N.x),0,main_frame$LO_INC_YR8_N.x) == 0, 0, 1))

main_frame$hi_size_all_6 <-
  ifelse(ifelse(is.na(main_frame$HI_INC_YR6_N.y),0,main_frame$HI_INC_YR6_N.y) == 0, 0,
  ifelse(ifelse(is.na(main_frame$HI_INC_YR6_N.x),0,main_frame$HI_INC_YR6_N.x) == 0, 0, 1))
main_frame$med_size_all_6 <-
  ifelse(ifelse(is.na(main_frame$MD_INC_YR6_N.y),0,main_frame$MD_INC_YR6_N.y) == 0, 0,
  ifelse(ifelse(is.na(main_frame$MD_INC_YR6_N.x),0,main_frame$MD_INC_YR6_N.x) == 0, 0, 1))
main_frame$low_size_all_6 <-
  ifelse(ifelse(is.na(main_frame$LO_INC_YR6_N.y),0,main_frame$LO_INC_YR6_N.y) == 0, 0,
  ifelse(ifelse(is.na(main_frame$LO_INC_YR6_N.x),0,main_frame$LO_INC_YR6_N.x) == 0, 0, 1))

main_frame$hi_size_all_4 <-
  ifelse(ifelse(is.na(main_frame$HI_INC_YR4_N.y),0,main_frame$HI_INC_YR4_N.y) == 0, 0,
  ifelse(ifelse(is.na(main_frame$HI_INC_YR4_N.x),0,main_frame$HI_INC_YR4_N.x) == 0, 0, 1))
main_frame$med_size_all_4 <-
  ifelse(ifelse(is.na(main_frame$MD_INC_YR4_N.y),0,main_frame$MD_INC_YR4_N.y) == 0, 0,
  ifelse(ifelse(is.na(main_frame$MD_INC_YR4_N.x),0,main_frame$MD_INC_YR4_N.x) == 0, 0, 1))
main_frame$low_size_all_4 <-
  ifelse(ifelse(is.na(main_frame$LO_INC_YR4_N.y),0,main_frame$LO_INC_YR4_N.y) == 0, 0,
  ifelse(ifelse(is.na(main_frame$LO_INC_YR4_N.x),0,main_frame$LO_INC_YR4_N.x) == 0, 0, 1))

main_frame$full_cohort <- ifelse(main_frame$low_size_all_4 == 1 & (main_frame$med_size_all_4 == 1 | mai

## Identify records with full completion information
main_frame$compall8 <- ifelse(main_frame$yr8.y == 0, 0, ifelse(main_frame$yr8.x == 0, 0, 1))
main_frame$compall6 <- ifelse(main_frame$yr6.y == 0, 0, ifelse(main_frame$yr6.x == 0, 0, 1))
main_frame$compall4 <- ifelse(main_frame$yr4.y == 0, 0, ifelse(main_frame$yr4.x == 0, 0, 1))

## Identify records with full cost information
main_frame$costall <- ifelse(is.na(main_frame$NPT41.y),0,ifelse(is.na(main_frame$NPT41.x),0,1))
main_frame$yr4school <- ifelse(main_frame$NPT41_PUB.x > 0 | main_frame$NPT41_PRIV.x > 0,1,0)

## Determine the change in completion rates
main_frame$low_comp_change <- main_frame$yr4.y - main_frame$yr4.x
main_frame$low_comp_change_rate <- (main_frame$yr4.y - main_frame$yr4.x) / main_frame$yr4.x
main_frame$low_cost_change <- main_frame$NPT41.y - main_frame$NPT41.x
main_frame$low_cost_change_rate <- (main_frame$NPT41.y - main_frame$NPT41.x)/main_frame$NPT41.x
main_frame$all_comp_change <- main_frame$COMP_ORIG_YR4_RT.y - main_frame$COMP_ORIG_YR4_RT.x
main_frame$all_cost_change <- main_frame$NPT4.y - main_frame$NPT4.x

main_frame$all_comp_change_rate <- (main_frame$COMP_ORIG_YR4_RT.y - main_frame$COMP_ORIG_YR4_RT.x)/ mai
main_frame$all_cost_change_rate <- (main_frame$NPT4.y - main_frame$NPT4.x) / main_frame$NPT4.x

main_frame$sat_change <- main_frame$SAT_AVG.y - main_frame$SAT_AVG.x
```

```
main_frame$adm_change <- main_frame$ADM_RATE.y - main_frame$ADM_RATE.x
main_frame$instrspend_change <- main_frame$INEXPFTE.y - main_frame$INEXPFTE.x
```

The final analysis_frame has 634 colleges and univeristies, down from 6706 initially, due to high amounts of missing data and eliminating about 1000 non bachelor's focused schools.

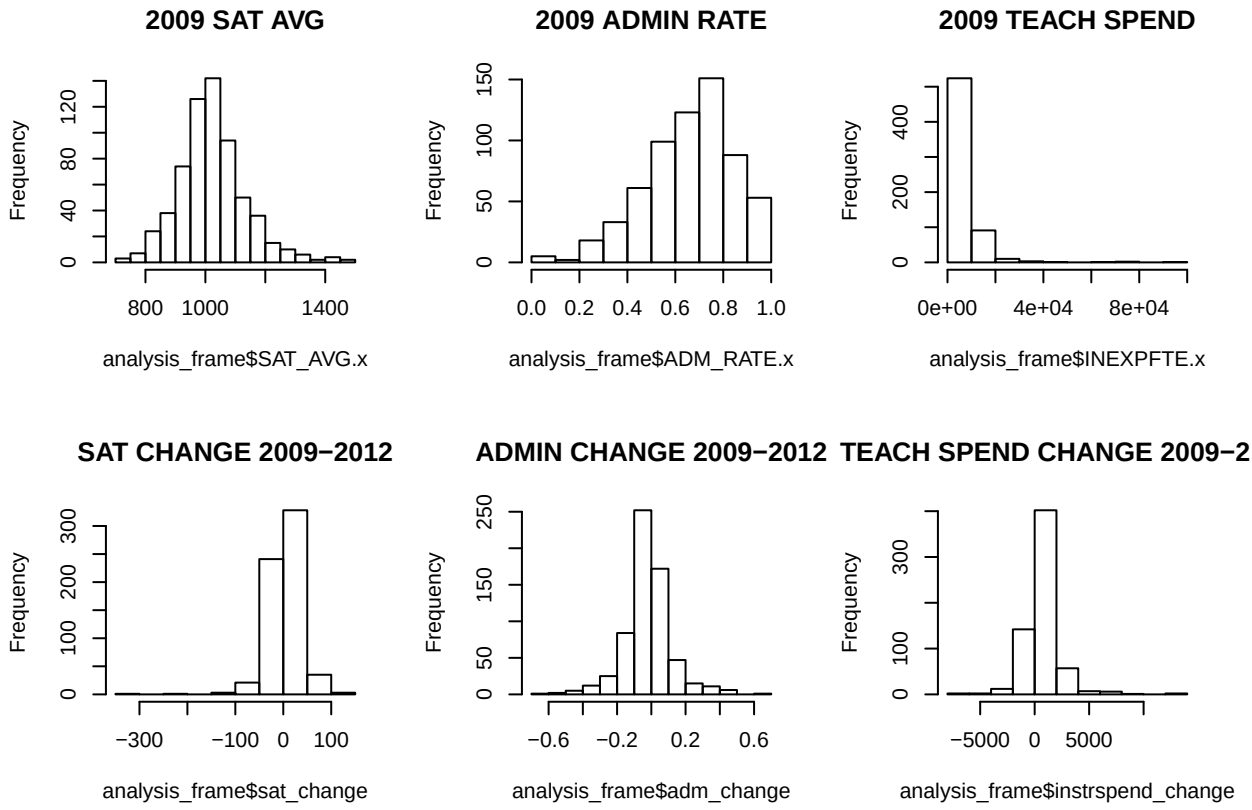
```
analysis_frame <-main_frame[
                              main_frame$full_cohort == 1
                             & main_frame$compall4 == 1
                             & main_frame$costall == 1
                             & main_frame$yr4school == 1
,c("UNITID","INSTNM.x","NPT41.y","NPT41.x","low_comp_change","low_cost_change","yr4.y","yr4.x","low_comp

analysis_frame <- analysis_frame[analysis_frame$PREDDEG.x==3,]
analysis_frame <- na.omit(analysis_frame)
analysis_frame <- analysis_frame[analysis_frame$all_cost_change > -20000,] ## (Cost went from very posi
```

Preliminary analysis:

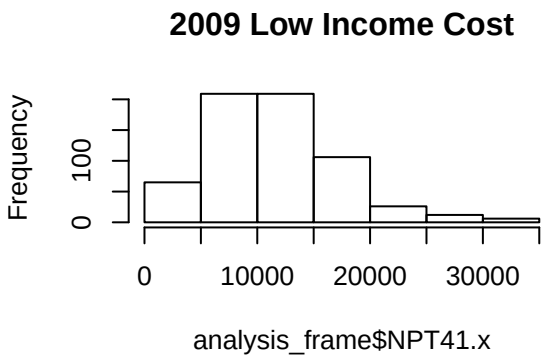
Control Variables

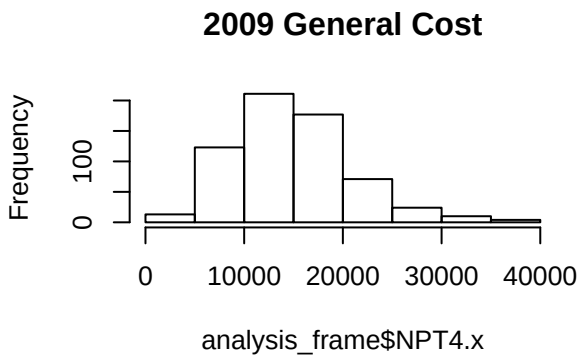
```
par(mfrow=c(2,3))
hist(analysis_frame$SAT_AVG.x, main="2009 SAT AVG")
hist(analysis_frame$ADM_RATE.x, main="2009 ADMIN RATE")
hist(analysis_frame$INEXPFTE.x, main="2009 TEACH SPEND") ##extreme outliers are ivy league schools
hist(analysis_frame$sat_change, main="SAT CHANGE 2009-2012")
hist(analysis_frame$adm_change, main="ADMIN CHANGE 2009-2012")
hist(analysis_frame$instrspend_change, main="TEACH SPEND CHANGE 2009-2012")
```

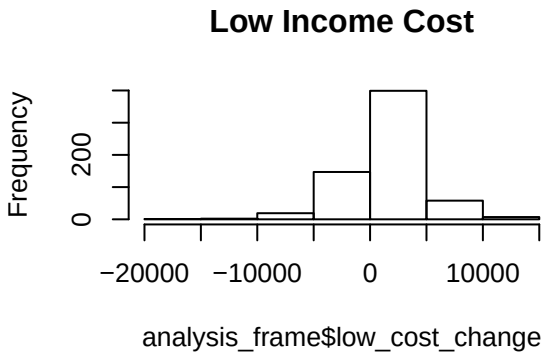


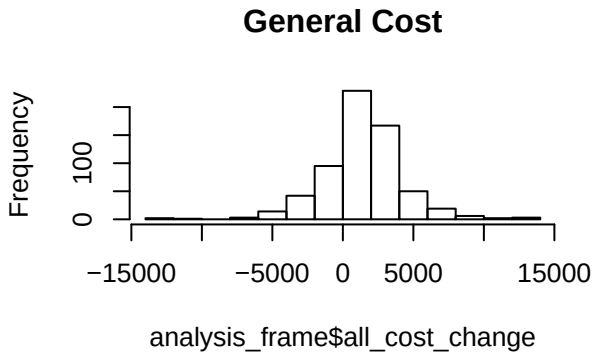
Dependent Variables

```
par(mfrow=c(2,2))
hist(analysis_frame$NPT41.x, main="2009 Low Income Cost")
hist(analysis_frame$NPT4.x, main="2009 General Cost")
hist(analysis_frame$low_cost_change, main="Low Income Cost")
hist(analysis_frame$all_cost_change, main="General Cost")
```



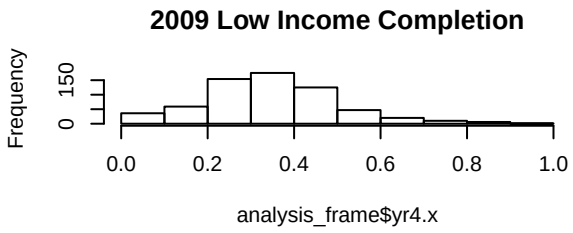


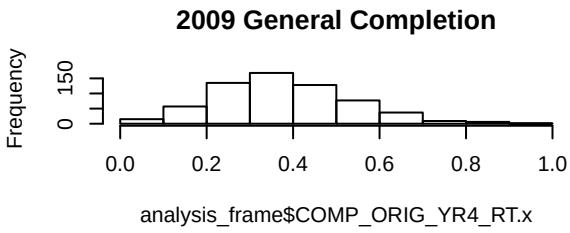


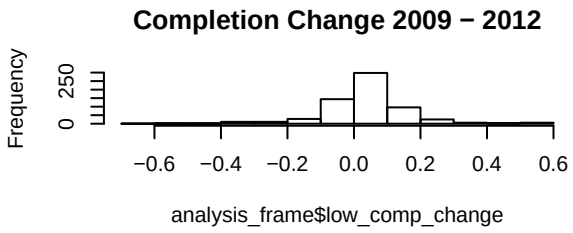


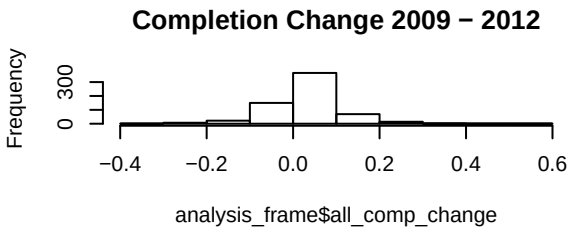
Independent Variables

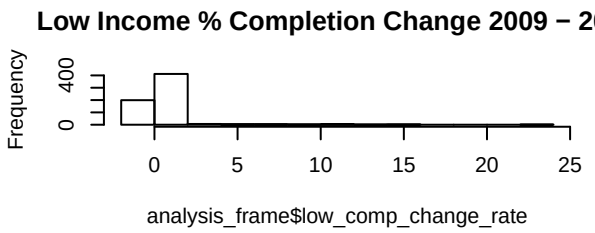
```
par(mfrow=c(3,2))
hist(analysis_frame$yr4.x, main="2009 Low Income Completion")
hist(analysis_frame$COMP_ORIG_YR4_RT.x, main="2009 General Completion")
hist(analysis_frame$low_comp_change, main="Completion Change 2009 - 2012")
hist(analysis_frame$all_comp_change, main="Completion Change 2009 - 2012")
hist(analysis_frame$low_comp_change_rate, main="Low Income % Completion Change 2009 - 2012")
hist(analysis_frame$all_comp_change_rate, main="All % Completion Change  2009 - 2012")
```

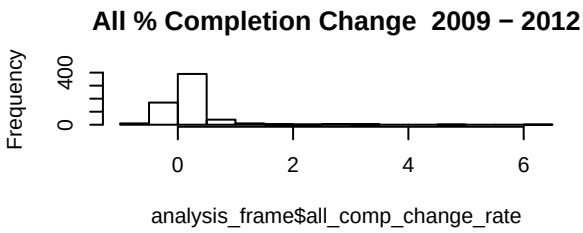




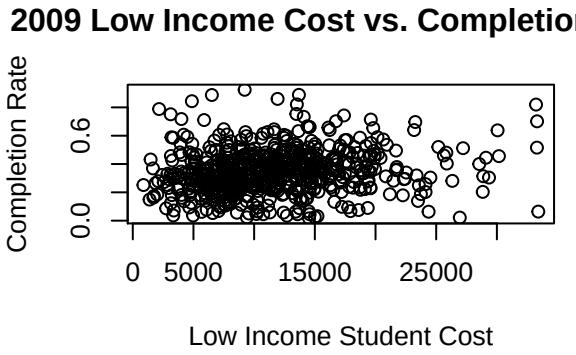


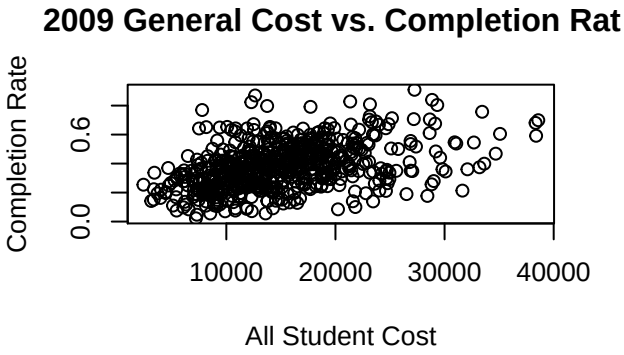


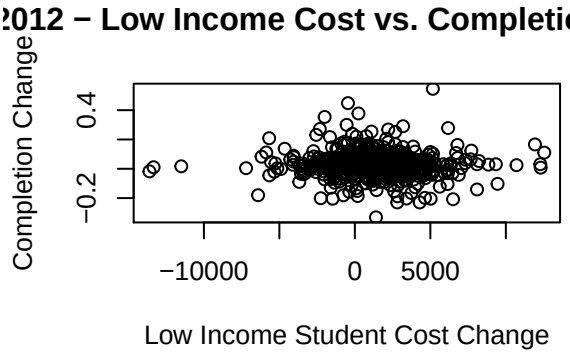


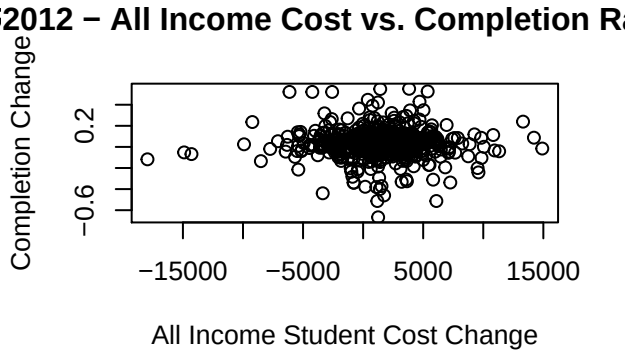



```
par(mfrow=c(2,2))
plot(analysis_frame$NPT41.x,analysis_frame$yr4.x, main = "2009 Low Income Cost vs. Completion Rate", xla
plot(analysis_frame$NPT4.x,analysis_frame$COMP_ORIG_YR4_RT.x, main = "2009 General Cost vs. Completion R
plot(analysis_frame$all_cost_change,analysis_frame$all_comp_change, main = "2009 - 2012 - Low Income Cos
plot(analysis_frame$low_cost_change,analysis_frame$low_comp_change, main = "2009 - 2012 - All Income Cos
```









The extreme differences between these two plots demonstrate the need for a first differences model to understand the impact of cost.

First we estimate a standard linear model, similar to the

$y = b0 + x1b1 + \ldots + xkbk + u$

```
all_2009 <- lm(COMP_ORIG_YR4_RT.x * 100 ~ NPT4.x + ADM_RATE.x + SAT_AVG.x + INEXPFTE.x, data = analysis
summary(all_2009)
```

```
##
## Call:
## lm(formula = COMP_ORIG_YR4_RT.x * 100 ~ NPT4.x + ADM_RATE.x +
##     SAT_AVG.x + INEXPFTE.x, data = analysis_frame)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -47.965  -7.361   0.797   7.465  32.241
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.410e+01  4.887e+00  -6.978 7.62e-12 ***
## NPT4.x       7.258e-04  7.658e-05   9.477  < 2e-16 ***
## ADM_RATE.x  -4.421e+00  2.635e+00  -1.677 0.093948 .
## SAT_AVG.x    6.012e-02  4.834e-03  12.437  < 2e-16 ***
## INEXPFTE.x   2.918e-04  7.973e-05   3.659 0.000274 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.38 on 628 degrees of freedom
## Multiple R-squared:  0.4554, Adjusted R-squared:  0.452
## F-statistic: 131.3 on 4 and 628 DF,  p-value: < 2.2e-16
```

```
low_2009 <- lm(yr4.x * 100 ~ NPT41.x + ADM_RATE.x + SAT_AVG.x + INEXPFTE.x, data = analysis_frame)
summary(low_2009)
```

```
##
## Call:
## lm(formula = yr4.x * 100 ~ NPT41.x + ADM_RATE.x + SAT_AVG.x +
##     INEXPFTE.x, data = analysis_frame)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -44.107  -7.218   1.052   8.555  37.864
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.054e+01  5.492e+00  -3.740 0.000201 ***
## NPT41.x      3.985e-04  8.889e-05   4.484 8.72e-06 ***
## ADM_RATE.x  -5.823e+00  2.936e+00  -1.983 0.047806 *
## SAT_AVG.x    4.887e-02  5.300e-03   9.222  < 2e-16 ***
## INEXPFTE.x   5.322e-04  8.860e-05   6.007 3.21e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.68 on 628 degrees of freedom
## Multiple R-squared:  0.3277, Adjusted R-squared:  0.3235
## F-statistic: 76.54 on 4 and 628 DF,  p-value: < 2.2e-16
```

However, it is very likely both of these models suffer from the same problem as the model before, unobserved variables:

$$y = b_0 + x_1 b_1 + ... + x_k b_k + c + u$$

If c is imagined as a school's general quality or the ernstwhile nature of the students it attracts, it is likely coorelated with the primary variable we care about, price. If this is a case, the covariance of the total error terms will not be zero.

Therefore, it is required that we utilized an unobserved effects model, with first differencing to remove the unobserved effects. This will isolate the variables of concern.

Supose a standard unobserved effects model:

$$y_{it} = x_{it} b + c_i + u_{it}$$

if you subtract out the previous within-unit observation, we have a first-difference transformation:

$$\Delta y_{it} = \Delta x b + \Delta * u_{it}$$

This model eliminates c, reducing the liklihood of the covaraiance of errors not being zero.

```r
all_change <- lm(all_comp_change * 100 ~ all_cost_change + sat_change + adm_change + instrspend_change,
summary(all_change)
```

```
##
## Call:
## lm(formula = all_comp_change * 100 ~ all_cost_change + sat_change +
##     adm_change + instrspend_change, data = analysis_frame)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -35.882  -4.091   0.250   3.687  51.232
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        3.2954163  0.4050547   8.136  2.2e-15 ***
## all_cost_change   -0.0001248  0.0001173  -1.064    0.288
## sat_change         0.0149901  0.0095578   1.568    0.117
## adm_change        -0.4427345  2.3874625  -0.185    0.853
## instrspend_change  0.0002103  0.0002028   1.037    0.300
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.305 on 628 degrees of freedom
## Multiple R-squared:  0.007651,   Adjusted R-squared:  0.001331
## F-statistic: 1.211 on 4 and 628 DF,  p-value: 0.3051
```

```r
low_change <- lm(low_comp_change * 100 ~ low_cost_change + sat_change + adm_change + instrspend_change,
summary(all_change)
```

```
##
## Call:
## lm(formula = all_comp_change * 100 ~ all_cost_change + sat_change +
##     adm_change + instrspend_change, data = analysis_frame)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -35.882  -4.091   0.250   3.687  51.232
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        3.2954163  0.4050547   8.136  2.2e-15 ***
## all_cost_change   -0.0001248  0.0001173  -1.064    0.288
## sat_change         0.0149901  0.0095578   1.568    0.117
## adm_change        -0.4427345  2.3874625  -0.185    0.853
## instrspend_change  0.0002103  0.0002028   1.037    0.300
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.305 on 628 degrees of freedom
## Multiple R-squared:  0.007651,   Adjusted R-squared:  0.001331
## F-statistic: 1.211 on 4 and 628 DF,  p-value: 0.3051
```

Here, the significance of all the variables disappears. This suggests it is unobserved factors about schools and students that drive completion rate rather than direct changes to the cost or price.

3 Part 2

We start by analyzing the effect of grants and loans on college student graduation rate.

3.1 Data Source

The data source used here is the Delta Cost Project Database made available by the Integrated Postsecondary Education Data System (IPEDS) [5]. This is a longitudinal study (panel data) from 1987 to 2012, consisting of 974 variables collected from postsecondary institutions throughout the United States. For the purposes of this study, we restrict ourselves to the cross sectional data from 2009 due to it having the best combination of completeness and recency.

3.2 Data Dictionary

- fed_grant_num: Number of full-time first-time degree/certificate-seeking undergraduates receiving federal grants
- fed_grant_pct: Percentage of full-time first-time degree/certificate-seeking undergraduates receiving federal grants
- fed_grant_avg_amount: Average amount of federal grants received by full-time first-time degree/certificate-seeking undergraduates
- state_grant_num: Number of full-time first-time degree/certificate-seeking undergraduates receiving state/local grants
- state_grant_pct: Percentage of full-time first-time degree/certificate-seeking undergraduates receiving state/local grants
- state_grant_avg_amount: Average amount of state/local grants received by full-time first-time degree/certificate-seeking undergraduates
- loan_num: Number of full-time first-time degree/certificate-seeking undergraduates receiving student loans
- loan_pct: Percentage of full-time first-time degree/certificate-seeking undergraduates receiving student loans
- loan_avg_amount: Average amount of student loans received by full-time first-time degree/certificate-seeking undergraduates
- bachelordegrees: Number of bachelor's degrees granted
- grad_rate_150_n: Number of students graduating within 150 percent of normal time
- grad_rate_150_p: Percentage of students graduating within 150 percent of normal time
- ftretention_rate: Full-time retention rate

3.3 Data Cleaning and EDA

Examine data without NA by following :

```
library(car)
library(stargazer)
library(sandwich)
library(zoo)
library(lmtest)
library(lattice)
```

```
library(survival)
library(Formula)
library(ggplot2)
library(Hmisc)
library(aod)
```

```
data_no_NA = delta_public_00_12 <- read.csv("~/R/Data/MyData_years.csv")
str(data_no_NA)
```

```
## 'data.frame':    14761 obs. of  15 variables:
##  $ X                     : int  1 13 217 349 1393 1394 1790 1792 1793 2026 ...
##  $ academicyear          : int  2012 2012 2012 2011 2011 2012 2009 2011 2012 2009 ...
##  $ fed_grant_num         : int  12 2556 242 8 61 61 41 103 79 15 ...
##  $ fed_grant_pct         : int  100 68 89 89 39 38 58 77 77 10 ...
##  $ fed_grant_avg_amount  : int  5205 3991 2811 4094 4766 4512 4681 4213 4325 2702 ...
##  $ state_grant_num       : int  6 63 3 7 8 5 1 6 2 5 ...
##  $ state_grant_pct       : int  50 2 1 78 5 3 1 4 2 3 ...
##  $ state_grant_avg_amount: int  1510 1031 7054 7377 2263 1205 13231 2418 9300 1599 ...
##  $ loan_num              : int  12 2339 221 1 102 111 66 105 66 17 ...
##  $ loan_pct              : int  100 62 82 11 66 69 93 78 65 12 ...
##  $ loan_avg_amount       : int  9000 5409 960 5500 6059 7211 14388 7001 7481 14477 ...
##  $ bachelordegrees       : num  20 407 46 22 37 58 160 272 312 40 ...
##  $ grad_rate_150_n       : num  13 132 73 2 29 45 7 12 2 230 ...
##  $ grad_rate_150_p       : num  0.265 0.208 0.427 0.5 0.509 ...
##  $ ftretention_rate      : num  1 0.18 0.38 0.9 0.61 0.68 0.28 0.18 0.63 0.88 ...
```

3.3.1 Cleaning `fed_grant_num`, `fed_grant_pct`, and `fed_grant_avg_amount`

Most (95%) observations of `fed_grant_pct` are between 0 and 82, which is what we expect for a percentage.

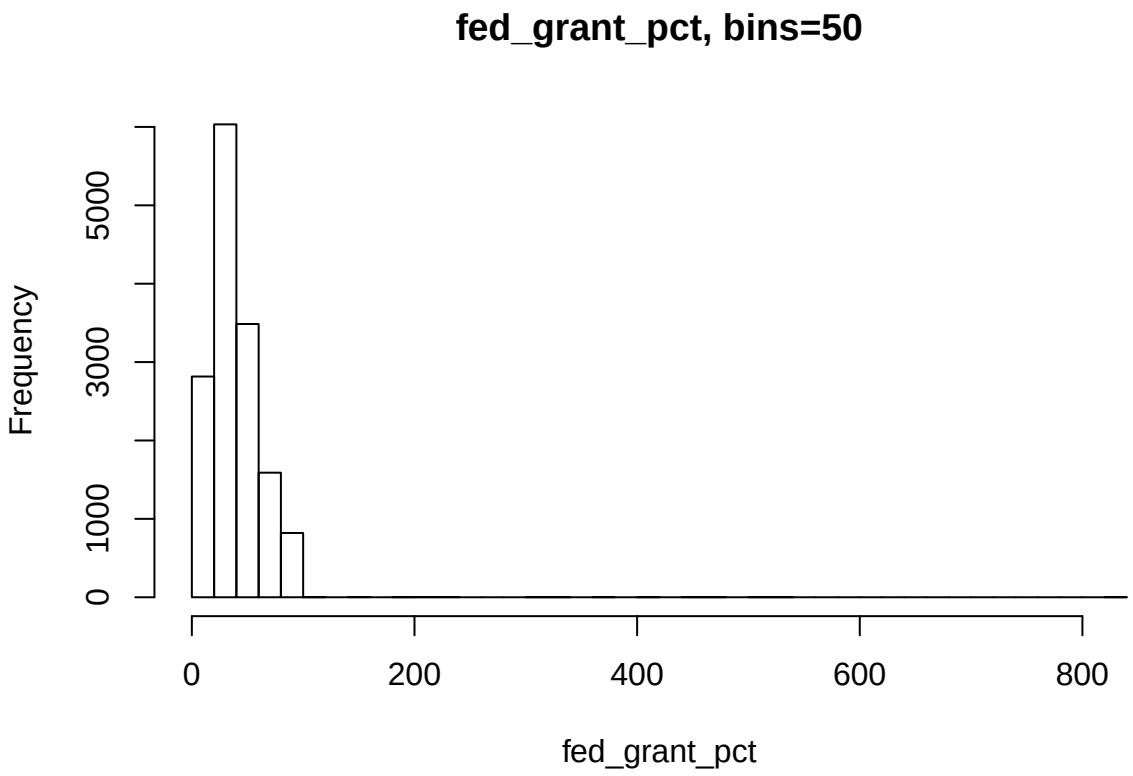
```
summary(data_no_NA$fed_grant_pct)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   23.00   35.00   39.57   51.00  830.00
```

```
describe(data_no_NA$fed_grant_pct)
```

```
## data_no_NA$fed_grant_pct
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##   14761       0     117       1   39.57      12      16      23      35
##     .75     .90     .95
##      51      71      82
##
## lowest :   1   2   3   4   5, highest: 455 476 506 534 830
```

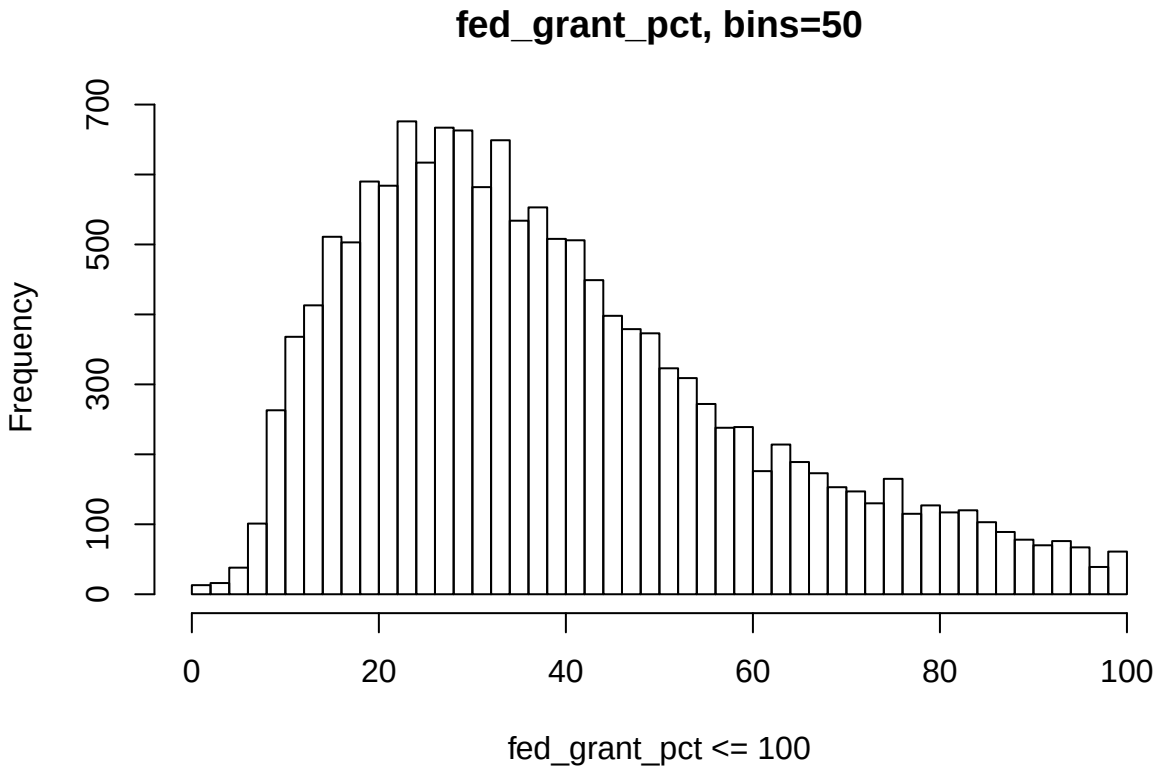
```
hist(data_no_NA$fed_grant_pct,
     main='fed_grant_pct, bins=50', xlab='fed_grant_pct', breaks=50)
```




```
length(data_no_NA$fed_grant_pct[data_no_NA$fed_grant_pct > 100])
```

```
## [1] 17
```

```
hist(data_no_NA$fed_grant_pct[data_no_NA$fed_grant_pct <= 100],
     main='fed_grant_pct, bins=50', xlab='fed_grant_pct <= 100', breaks=50)
```



17 observations of `fed_grant_pct` are greater than 100, ranging from 103 to 830. One possible explanation for these is that they should be divided by 10. For now, we intend to simply omit these since there are so few.

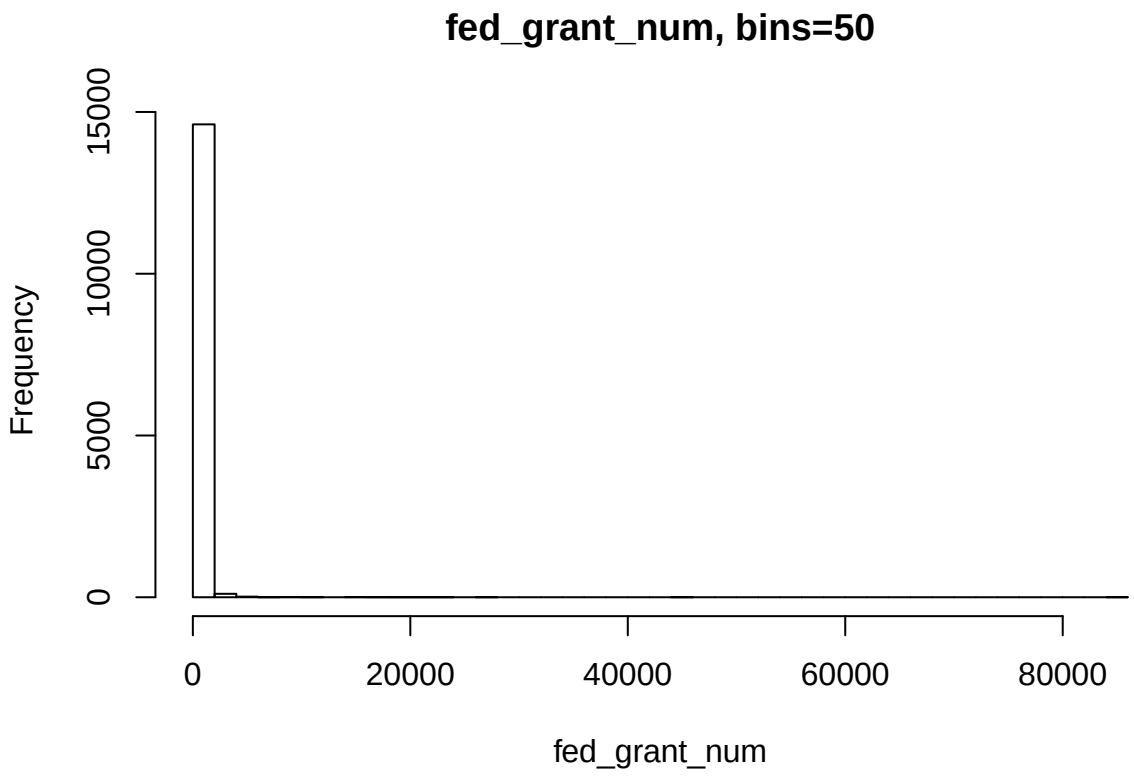
```
summary(data_no_NA$fed_grant_num)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     1.0    70.0   137.0   298.3   316.0 85070.0
```

```
describe(data_no_NA$fed_grant_num)
```

```
## data_no_NA$fed_grant_num
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##   14761       0    1479       1   298.3      17      33      70     137
##     .75     .90     .95
##     316     661     965
##
## lowest :     1     2     3     4     5
## highest: 21643 23439 26918 44268 85068
```

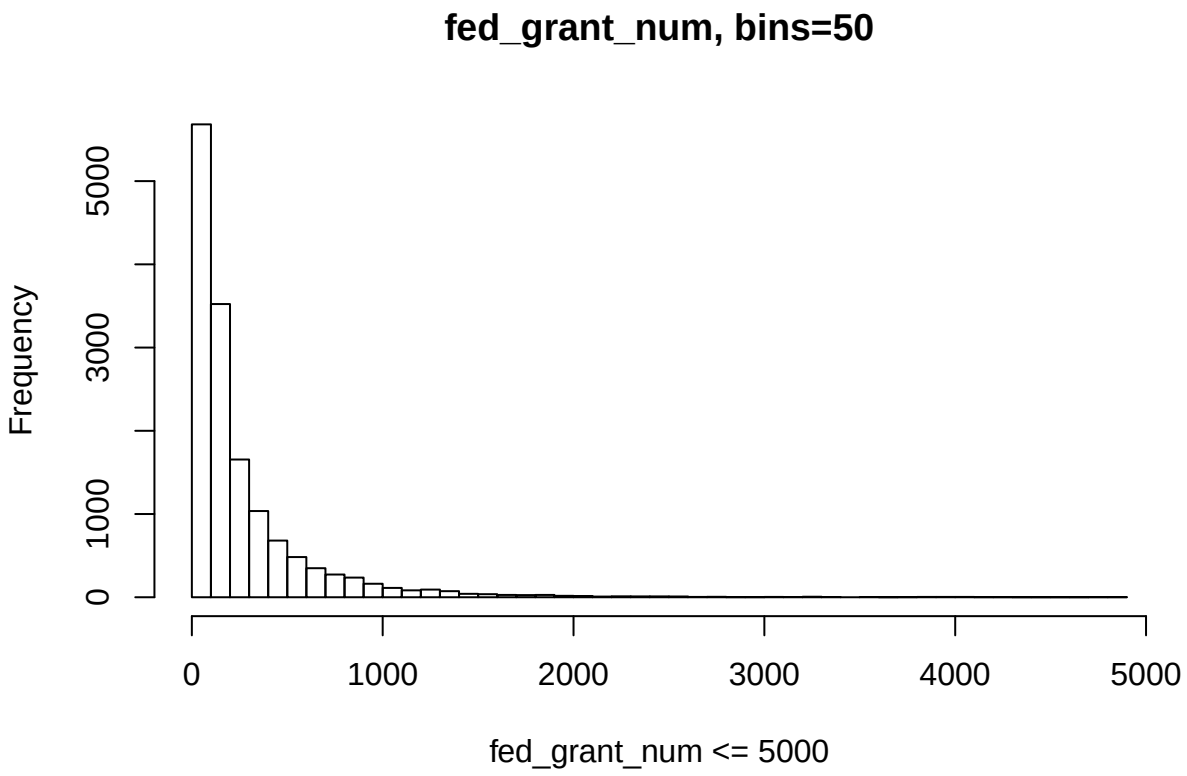
```
hist(data_no_NA$fed_grant_num,
     main='fed_grant_num, bins=50', xlab='fed_grant_num', breaks=50)
```




```
length(data_no_NA$fed_grant_num[data_no_NA$fed_grant_num > 5000])
```

```
## [1] 22
```

```
hist(data_no_NA$fed_grant_num[data_no_NA$fed_grant_num <= 5000],
     main='fed_grant_num, bins=50', xlab='fed_grant_num <= 5000', breaks=50)
```




```
length(data_no_NA$fed_grant_num[data_no_NA$fed_grant_num%%1 != 0])
```

```
## [1] 0
```

The 22 outliers of `fed_grant_num` greater than 5000 will be omitted. All of the observations are whole numbers, which is what we expect since it is a counting variable.

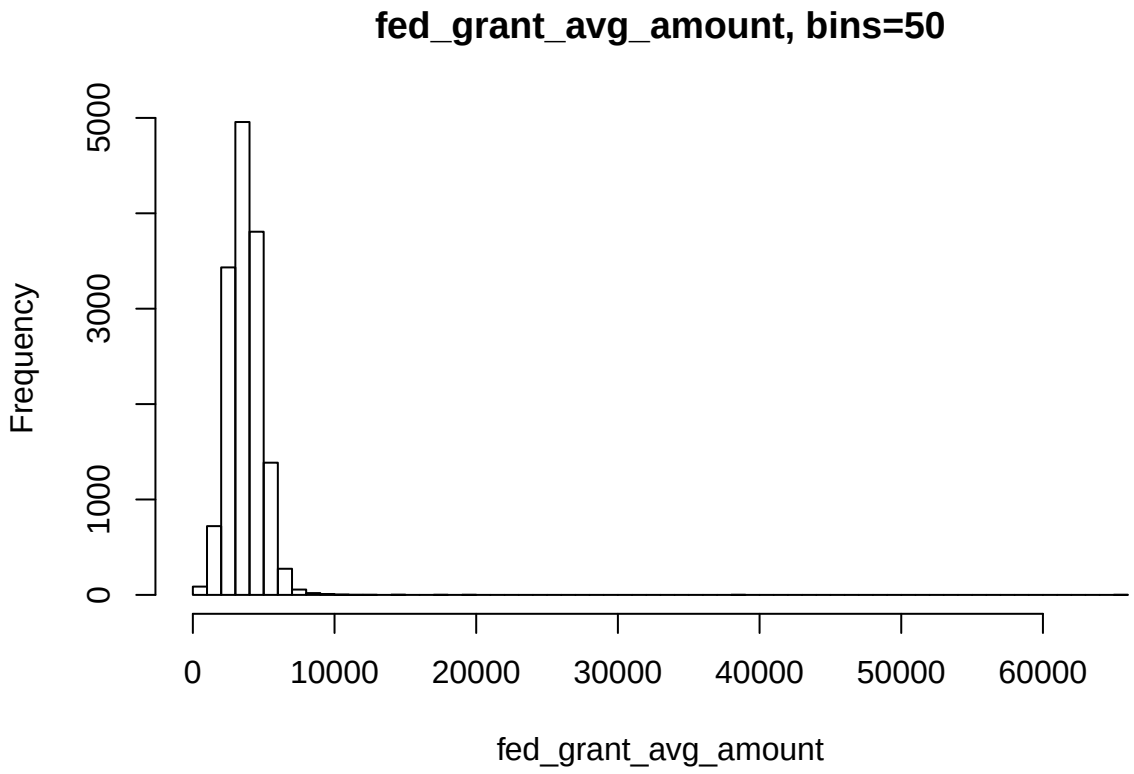
```
summary(data_no_NA$fed_grant_avg_amount)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     118    2913    3575    3698    4447   65600
```

```
describe(data_no_NA$fed_grant_avg_amount)
```

```
## data_no_NA$fed_grant_avg_amount
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##   14761       0    4413       1    3698    1967    2426    2913    3575
##     .75     .90     .95
##    4447    5108    5550
##
## lowest :   118   120   125   188   213
## highest: 14899 17551 19296 38786 65603
```

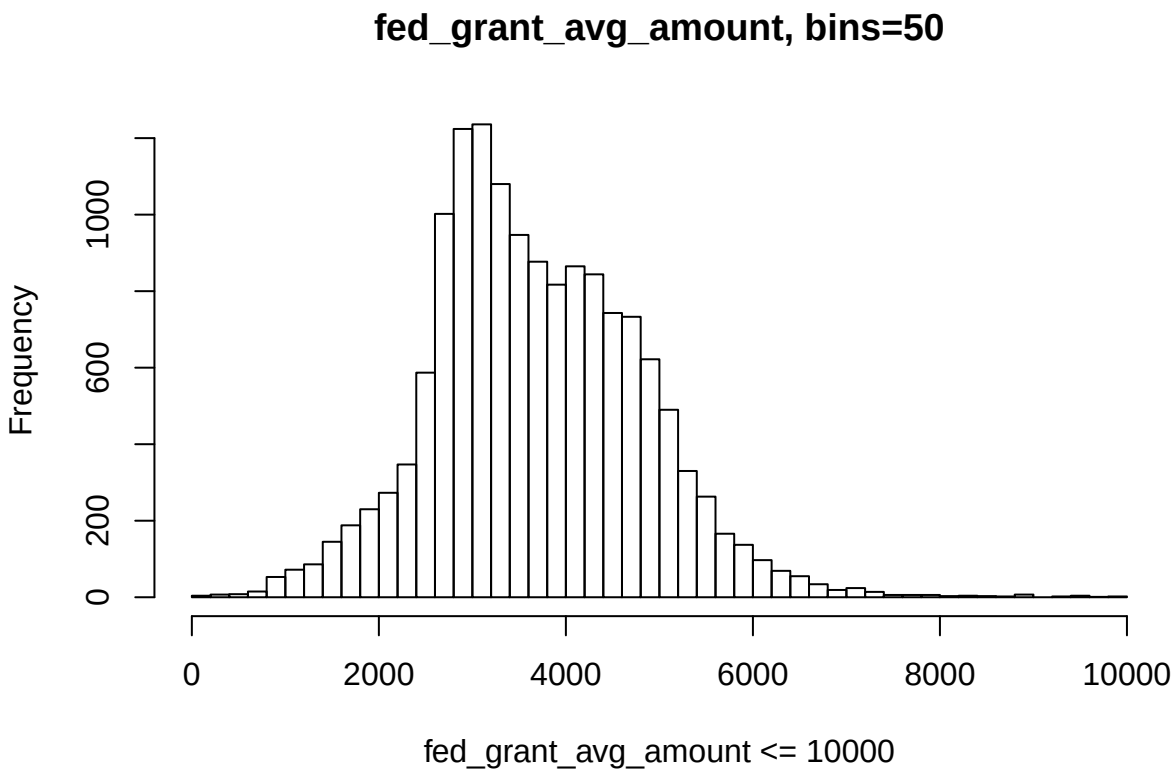
```r
hist(data_no_NA$fed_grant_avg_amount,
     main='fed_grant_avg_amount, bins=50', xlab='fed_grant_avg_amount', breaks=50)
```




```r
length(data_no_NA$fed_grant_avg_amount[data_no_NA$fed_grant_avg_amount > 10000])
```

```
## [1] 12
```

```r
hist(data_no_NA$fed_grant_avg_amount[data_no_NA$fed_grant_avg_amount <= 10000],
     main='fed_grant_avg_amount, bins=50', xlab='fed_grant_avg_amount <= 10000', breaks=50)
```



The 12 outliers of `fed_grant_avg_amount` greater than $10,000 will be omitted.

3.3.2 Cleaning `state_grant_num`, `state_grant_pct`, and `state_grant_avg_amount`

Most observations of `state_grant_pct` are between 0 and 100, but some are greater than 100.

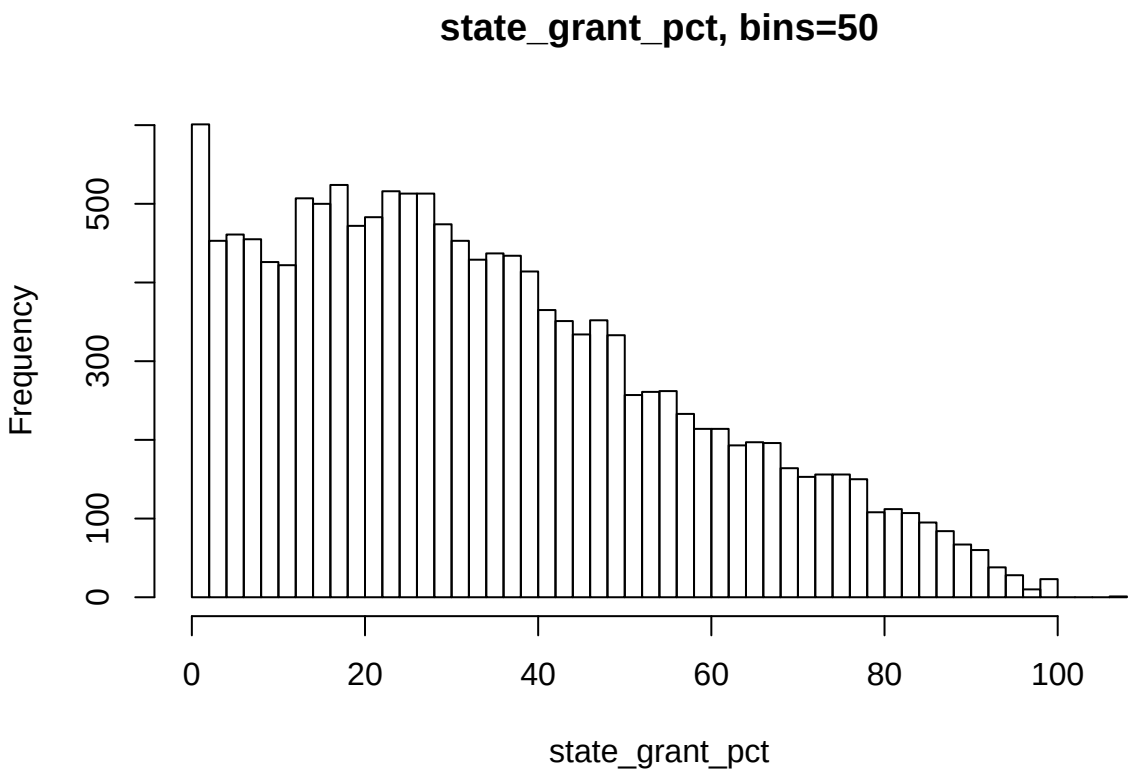
```
summary(data_no_NA$state_grant_pct)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   16.00   31.00   34.49   50.00  108.00
```

```
describe(data_no_NA$state_grant_pct)
```

```
## data_no_NA$state_grant_pct
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##   14761       0     102       1   34.49       3       6      16      31
##     .75     .90     .95
##      50      69      78
##
## lowest :   0   1   2   3   4, highest:  97  98  99 100 108
```

```
hist(data_no_NA$state_grant_pct, main='state_grant_pct, bins=50',
     xlab='state_grant_pct', breaks=50)
```




```
length(data_no_NA$state_grant_pct[data_no_NA$state_grant_pct > 100])
```

```
## [1] 1
```

The 1 observation of `state_grant_pct` greater than 100 will be omitted.

Some observations of `state_grant_num` are extreme outliers. The 95th percentile is 1068. There are 121 observations greater than 3000 and 55 greater than 5000.

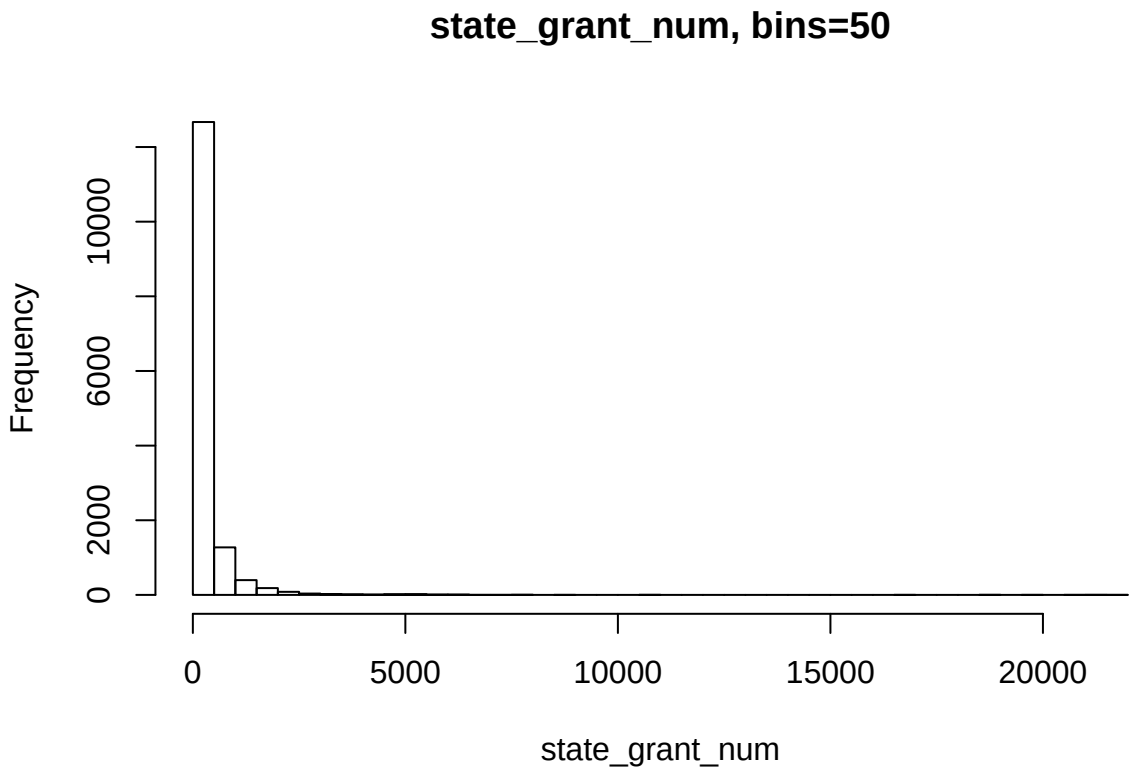
```
summary(data_no_NA$state_grant_num)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     1.0    43.0   121.0   291.5   282.0 21750.0
```

```
describe(data_no_NA$state_grant_num)
```

```
## data_no_NA$state_grant_num
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##   14761       0    1596       1   291.5       5      11      43     121
##     .75     .90     .95
##     282     665    1068
##
## lowest :     1     2     3     4     5
## highest: 19712 20644 21099 21486 21751
```

```r
hist(data_no_NA$state_grant_num, main='state_grant_num, bins=50',
     xlab='state_grant_num', breaks=50)
```




```r
length(data_no_NA$state_grant_num[data_no_NA$state_grant_num%%1 != 0])
```

```
## [1] 0
```

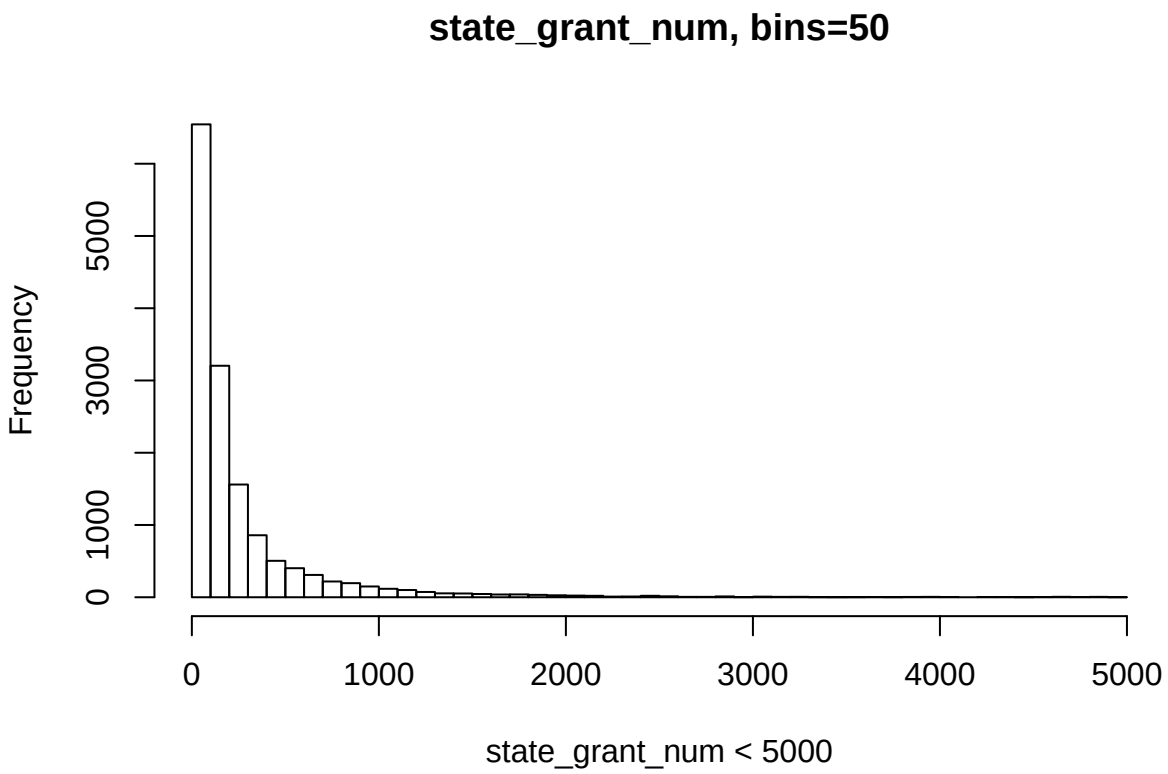
```r
length(data_no_NA$state_grant_num[data_no_NA$state_grant_num >= 3000])
```

```
## [1] 121
```

```r
length(data_no_NA$state_grant_num[data_no_NA$state_grant_num >= 5000])
```

```
## [1] 55
```

```r
hist(data_no_NA$state_grant_num[data_no_NA$state_grant_num < 5000],
     main='state_grant_num, bins=50', xlab='state_grant_num < 5000', breaks=50)
```



There are no observations of of `state_grant_num` that are not whole numbers, which is what we expect since it is a counting variable. Observations of `state_grant_num` greater than 5000 will be omitted due to being outliers.

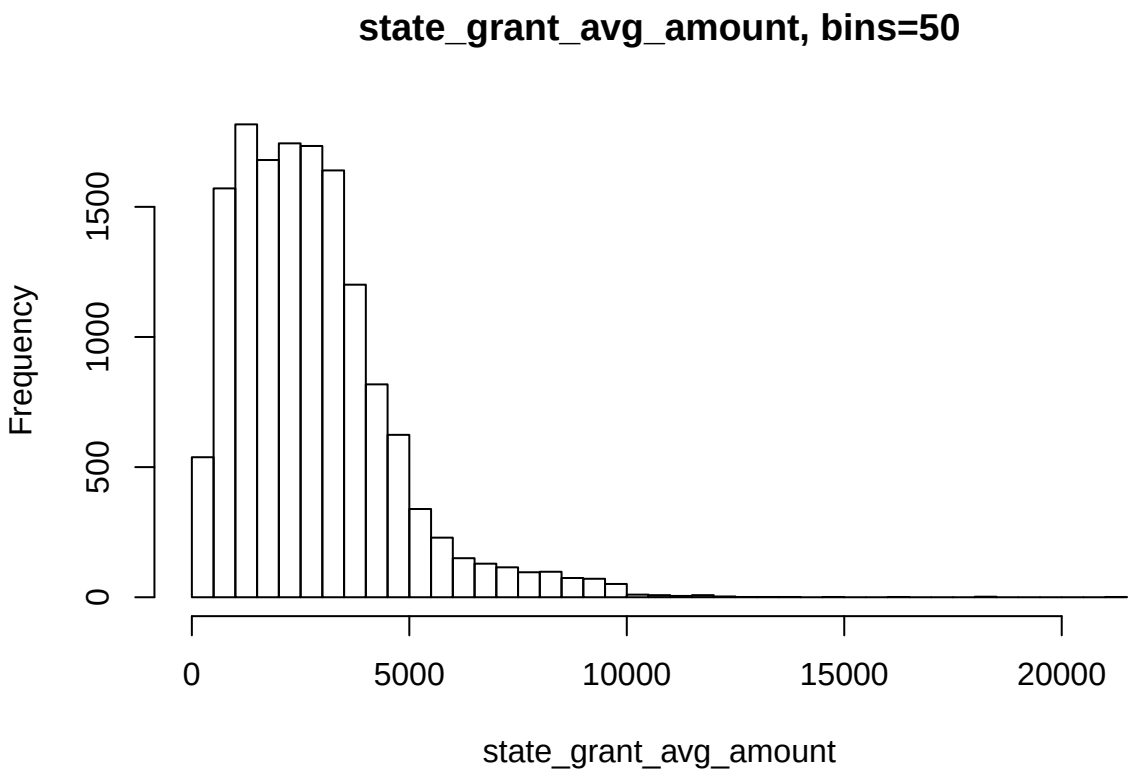
```
summary(data_no_NA$state_grant_avg_amount)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      30    1438    2511    2789    3635   21110
```

```
describe(data_no_NA$state_grant_avg_amount)
```

```
## data_no_NA$state_grant_avg_amount
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##   14761       0    5473       1    2789     595     855    1438    2511
##     .75     .90     .95
##    3635    4926    6259
##
## lowest :    30    50    80    83   100
## highest: 15000 16484 18322 18494 21106
```

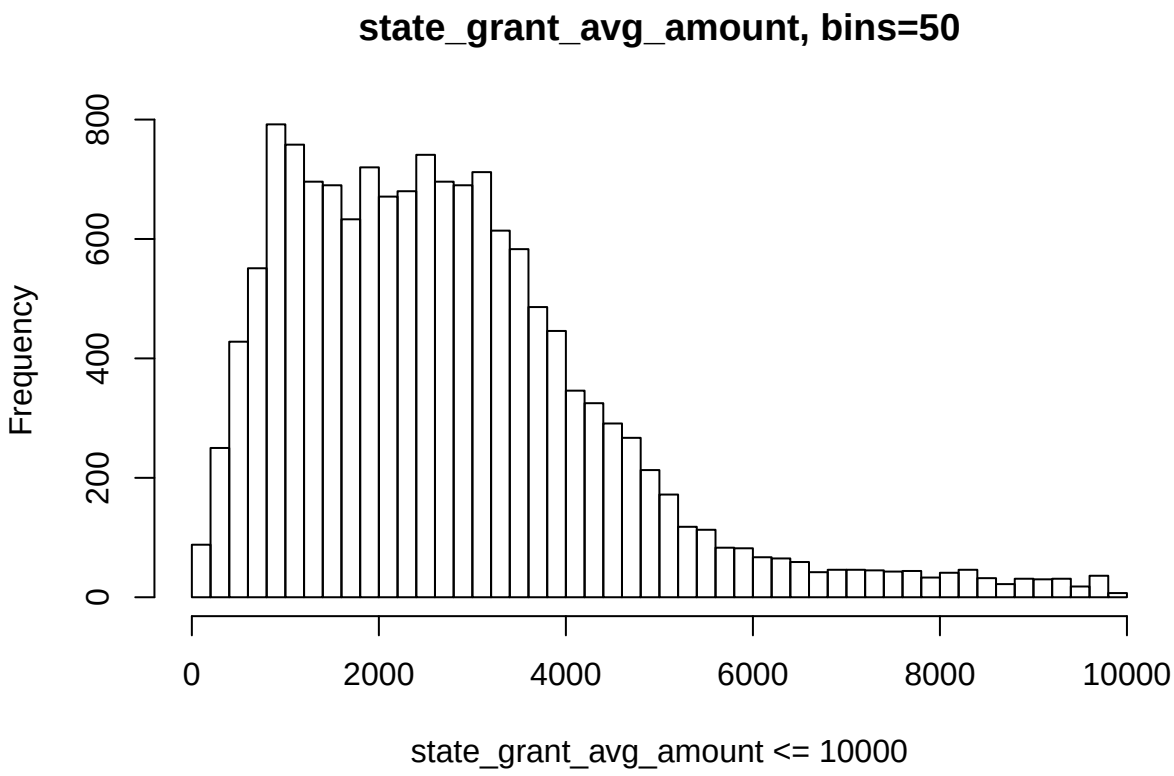
```
hist(data_no_NA$state_grant_avg_amount,
     main='state_grant_avg_amount, bins=50', xlab='state_grant_avg_amount', breaks=50)
```




```
length(data_no_NA$state_grant_avg_amount[data_no_NA$state_grant_avg_amount > 10000])
```

```
## [1] 42
```

```
hist(data_no_NA$state_grant_avg_amount[data_no_NA$state_grant_avg_amount <= 10000],
     main='state_grant_avg_amount, bins=50', xlab='state_grant_avg_amount <= 10000', breaks=50)
```



`state_grant_avg_amount` has 42 outliers greater than $10,000. Those observations will be omitted.

3.3.3 Cleaning `loan_num`, `loan_pct`, and `loan_avg_amount`

There are 17 anomalous rows where `loan_pct` is above 100.

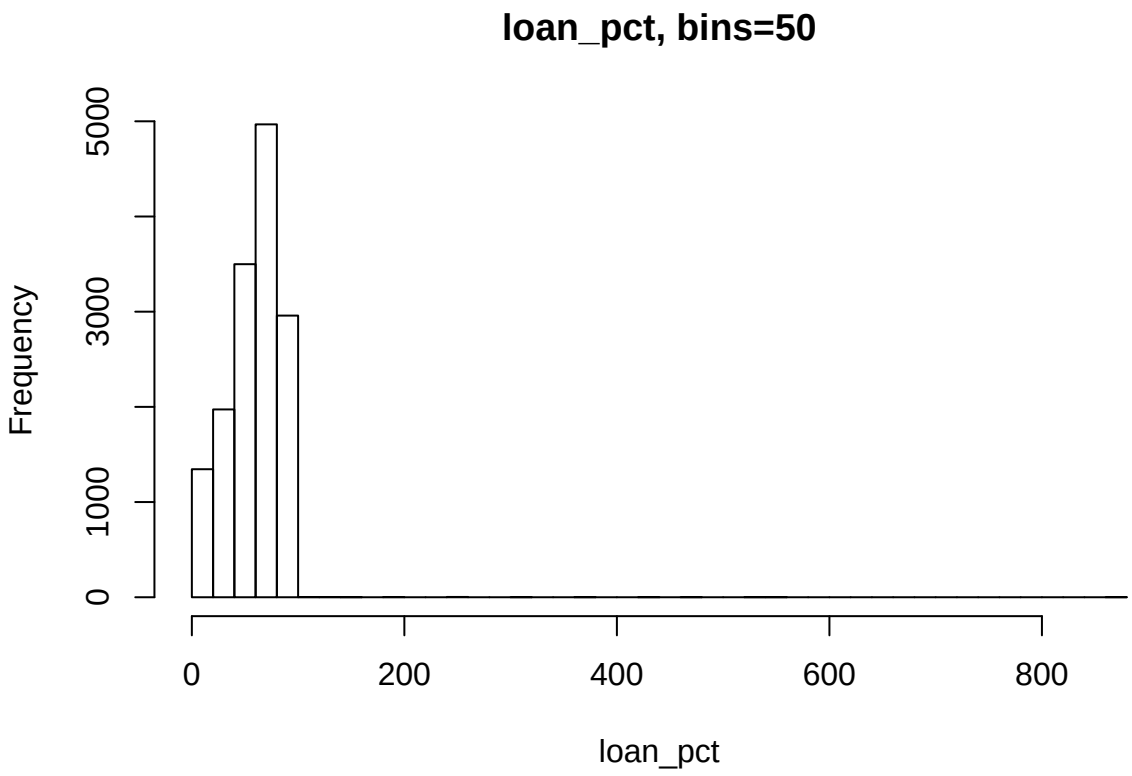
```
summary(data_no_NA$loan_pct)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   43.00   63.00   59.18   78.00  876.00
```

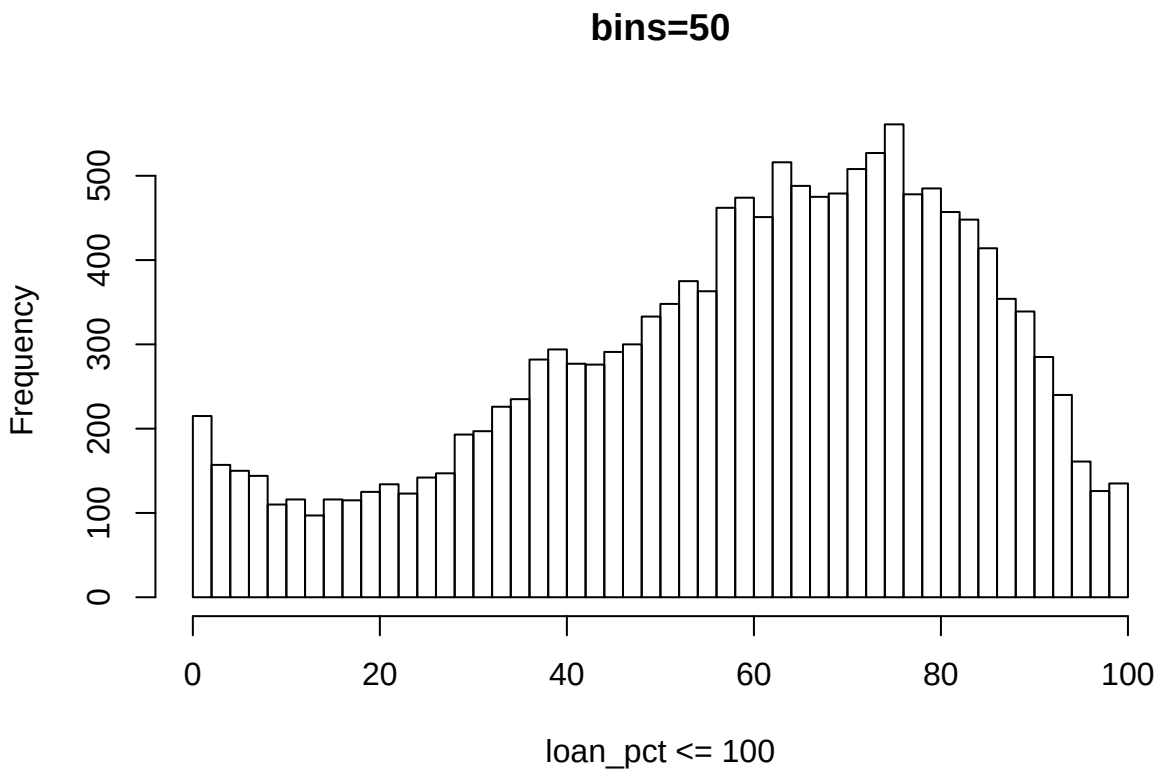
```
describe(data_no_NA$loan_pct)
```

```
## data_no_NA$loan_pct
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##   14761       0     118       1   59.18      10      22      43      63
##     .75     .90     .95
##      78      88      92
##
## lowest :   0   1   2   3   4, highest: 422 477 536 544 876
```

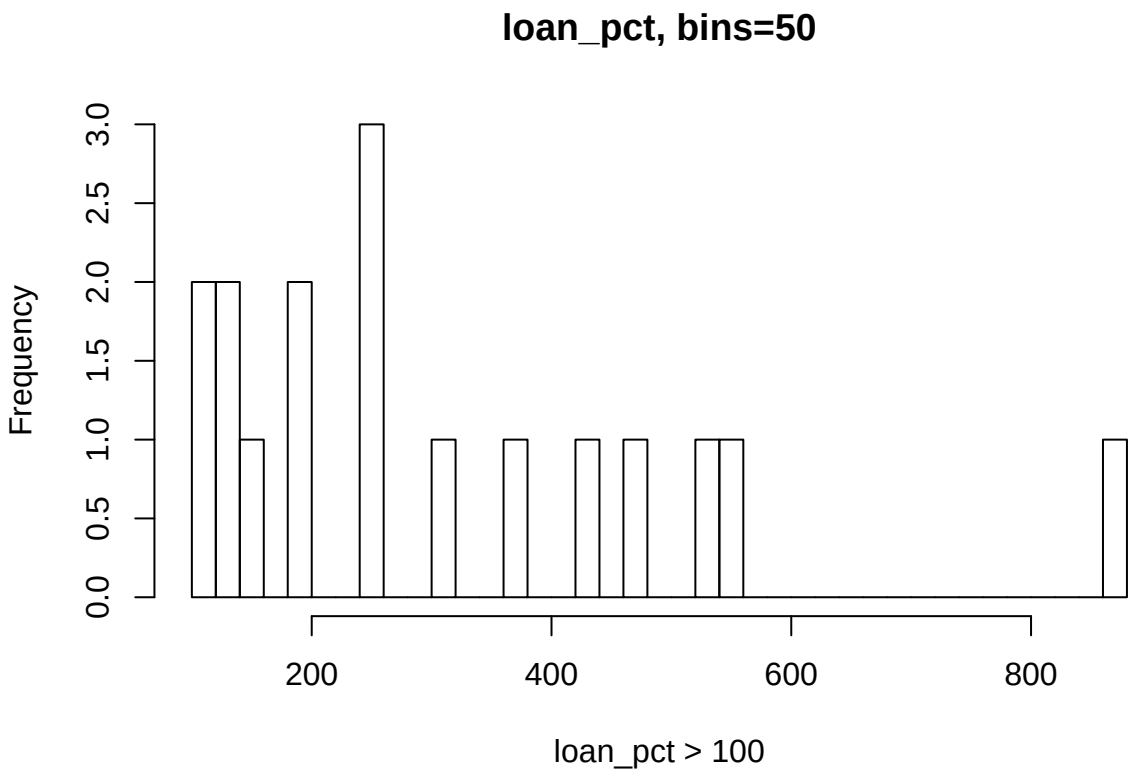
```
hist(data_no_NA$loan_pct, main='loan_pct, bins=50', xlab='loan_pct',
     breaks=50)
```




```
hist(data_no_NA$loan_pct[data_no_NA$loan_pct <= 100],
     main='bins=50', xlab='loan_pct <= 100', breaks=50)
```




```
hist(data_no_NA$loan_pct[data_no_NA$loan_pct > 100],
     main='loan_pct, bins=50', xlab='loan_pct > 100', breaks=50)
```




```
length(data_no_NA$loan_pct[data_no_NA$loan_pct > 100])
```

```
## [1] 17
```

The 17 outliers for `loan_pct` will be omitted.

There are also some extreme outliers for `loan_num` that are worth exploring.

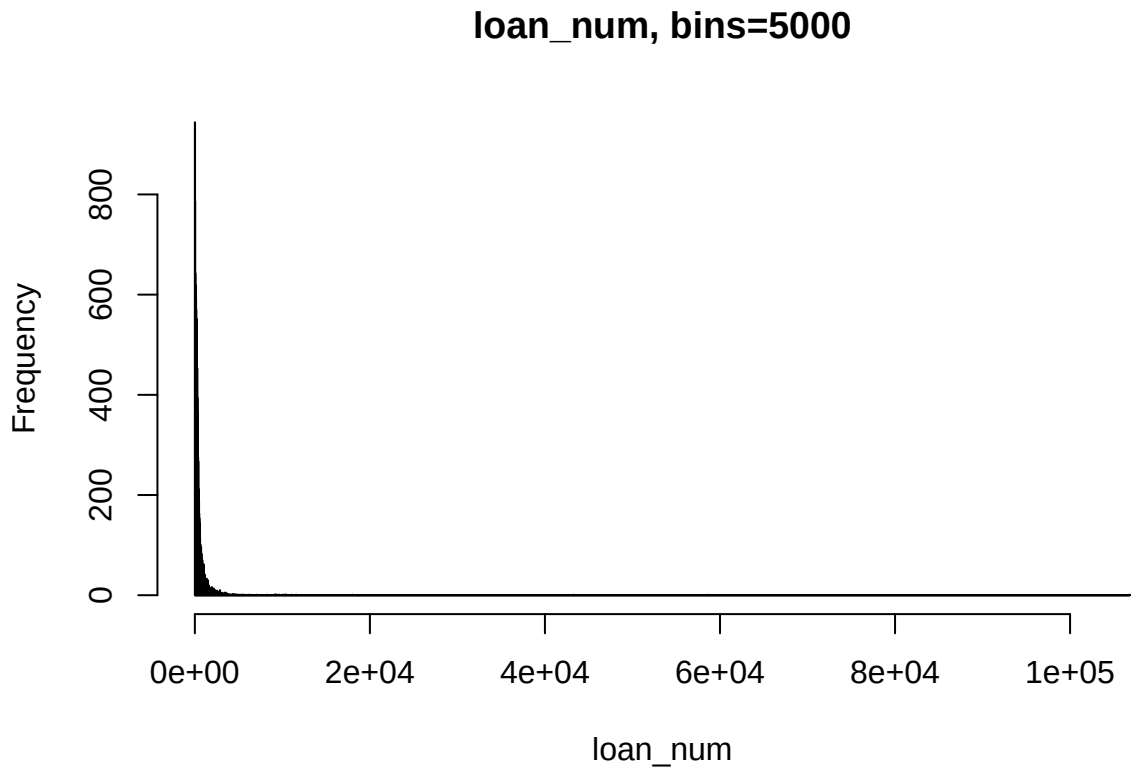
```
summary(data_no_NA$loan_num)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##      1.0    104.0    238.0    443.5    486.0 106800.0
```

```
describe(data_no_NA$loan_num)
```

```
## data_no_NA$loan_num
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##   14761       0    1980       1   443.5      16      33     104     238
##     .75     .90     .95
##     486    1001    1505
##
## lowest :      1      2      3      4      5
## highest:  11570  17999  43307  58606 106840
```

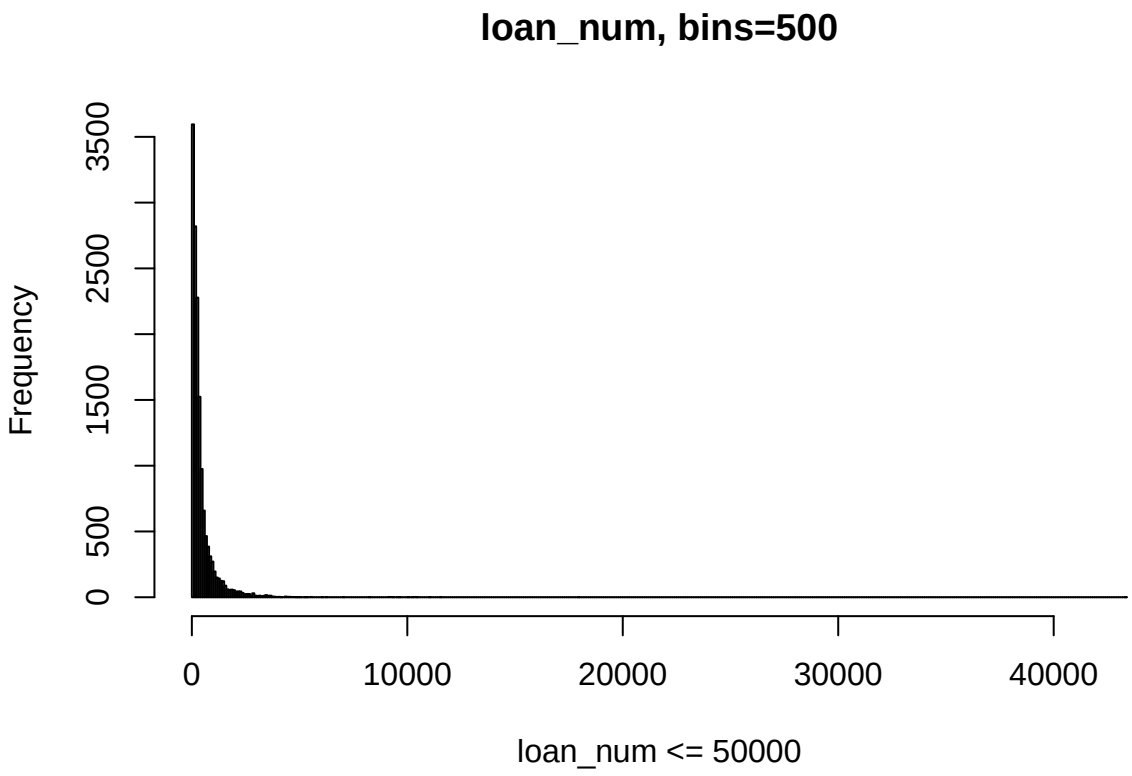
```
hist(data_no_NA$loan_num, main='loan_num, bins=5000', xlab='loan_num',
     breaks=5000)
```




```
length(data_no_NA$loan_num[data_no_NA$loan_num > 50000])
```

```
## [1] 2
```

```
hist(data_no_NA$loan_num[data_no_NA$loan_num <= 50000],
     main='loan_num, bins=500', xlab='loan_num <= 50000',
     breaks=500)
```




```
length(data_no_NA$loan_num[data_no_NA$loan_num%%1 != 0])
```

```
## [1] 0
```

The 2 rows where `loan_num` is greater than 50,000 will be omitted, since those are implausible numbers. All rows are whole numbers, which is what we expect since this is a counting variable.

Most of the `loan_avg_amount` observations are less than $7,000, but there are some high outliers on the right tail of the distribution.

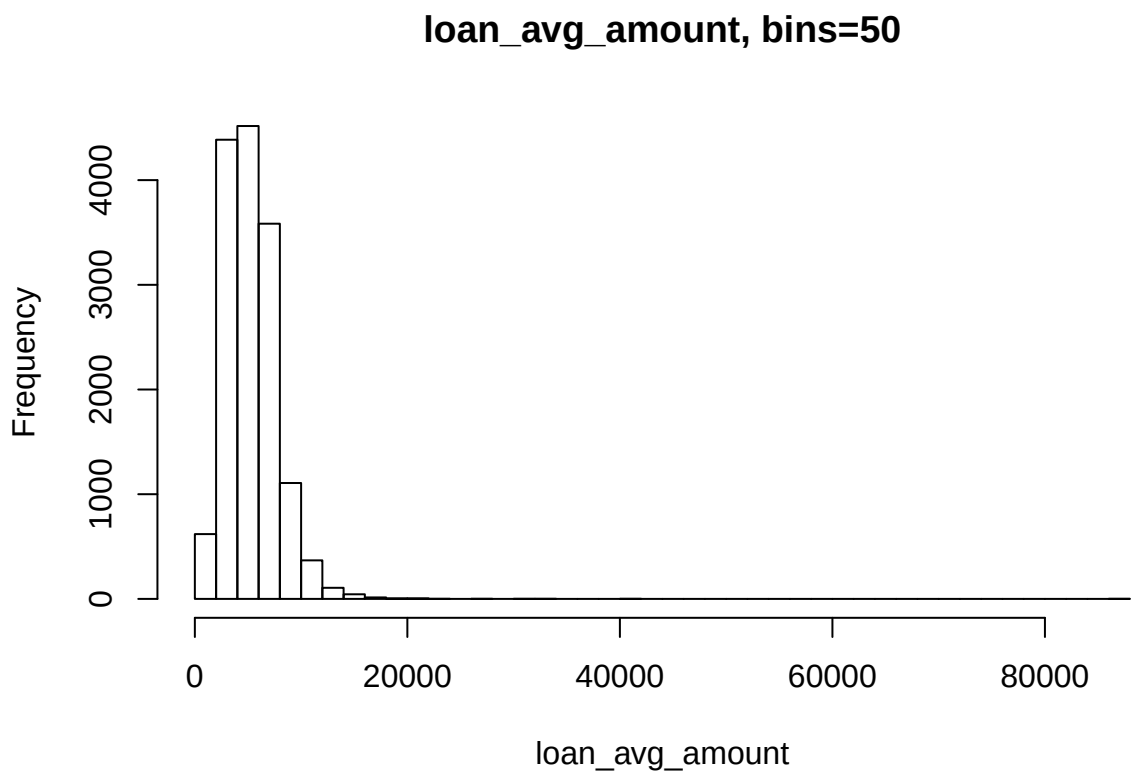
```
summary(data_no_NA$loan_avg_amount)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     100    3500    5054    5290    6668   86970
```

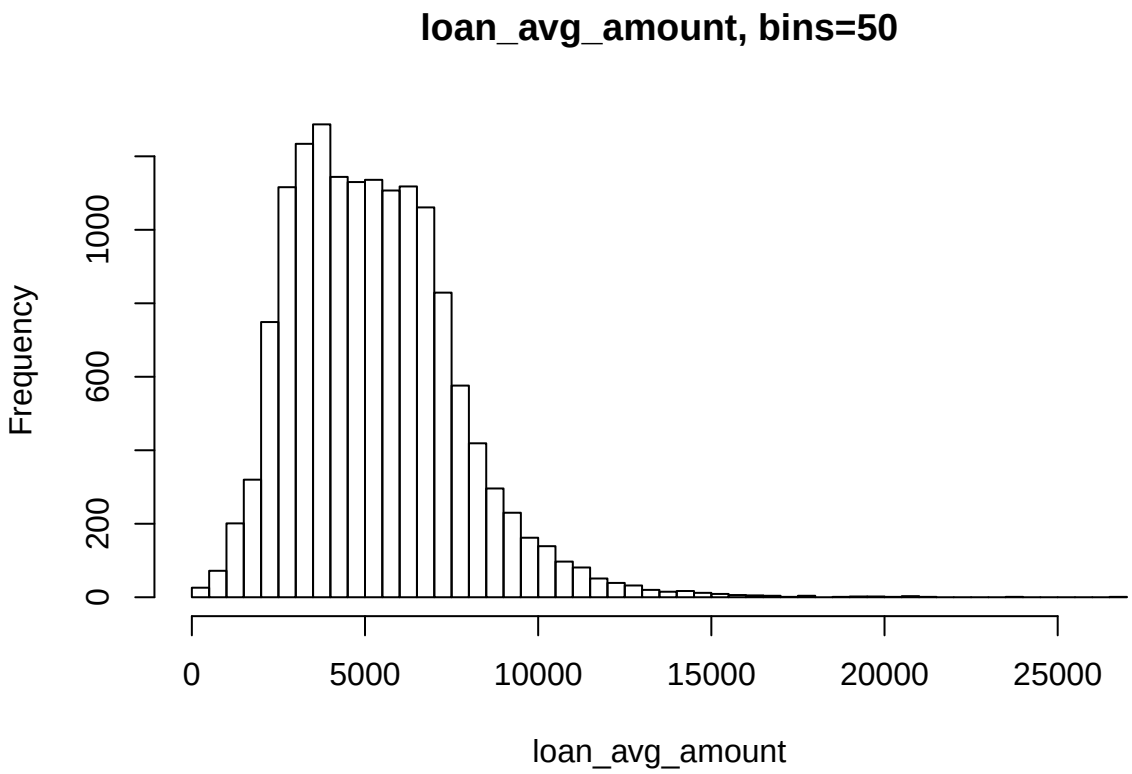
```
describe(data_no_NA$loan_avg_amount)
```

```
## data_no_NA$loan_avg_amount
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##   14761       0    6933       1    5290    2100    2575    3500    5054
##     .75     .90     .95
##    6668    8197    9427
##
## lowest :   100   102   112   121   125
## highest: 26827 31119 33105 41010 86971
```

```r
hist(data_no_NA$loan_avg_amount, main='loan_avg_amount, bins=50',
     xlab='loan_avg_amount', breaks=50)
```




```r
hist(data_no_NA$loan_avg_amount[data_no_NA$loan_avg_amount <= 30000],
     main='loan_avg_amount, bins=50', xlab='loan_avg_amount', breaks=50)
```




```
length(data_no_NA$loan_avg_amount[data_no_NA$loan_avg_amount > 30000])
```

```
## [1] 4
```

The 4 observations greater than $30,000 will be omitted.

3.3.4 Cleaning `bachelordegrees`

Analysis of `bachelordegrees` shows some extreme values.

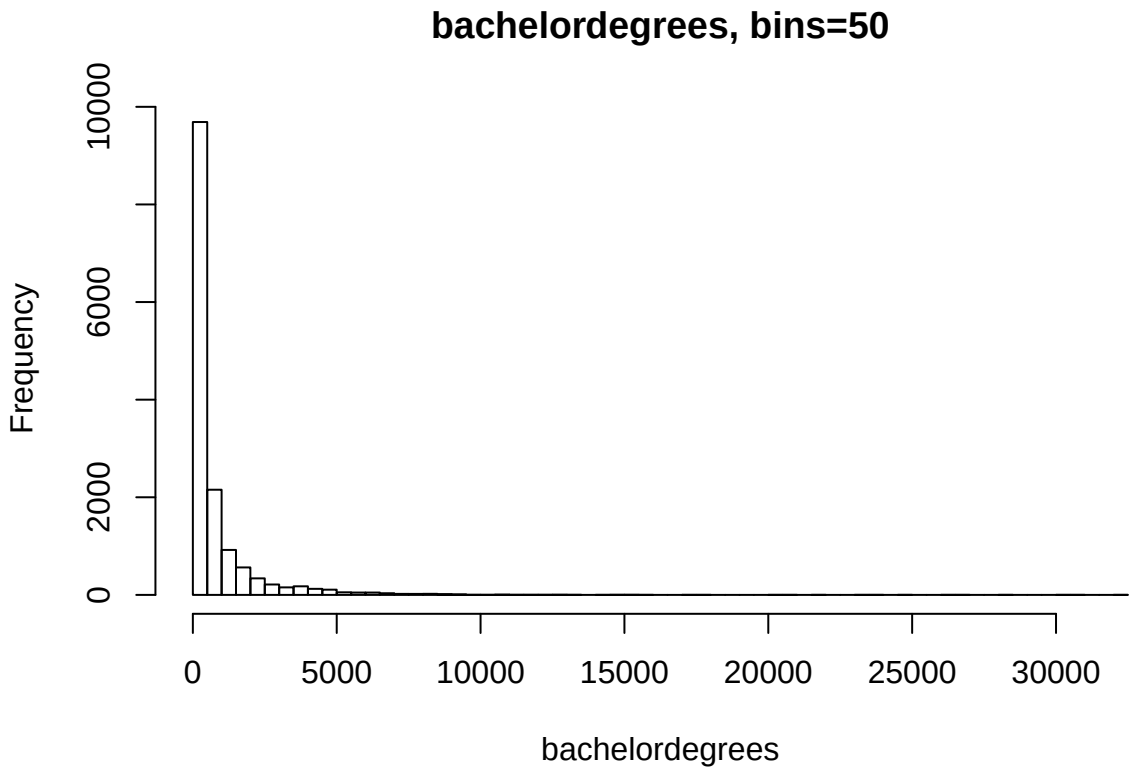
```
summary(data_no_NA$bachelordegrees)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0    64.0   294.0   765.4   737.0 32430.0
```

```
describe(data_no_NA$bachelordegrees)
```

```
## data_no_NA$bachelordegrees
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##   14761       0    2969    0.99   765.4       0       0      64     294
##     .75     .90     .95
##     737    1929    3464
##
## lowest :     0     1     2     3     4
## highest: 26805 28060 30032 30882 32432
```

```r
hist(data_no_NA$bachelordegrees,
     main='bachelordegrees, bins=50', xlab='bachelordegrees', breaks=50)
```



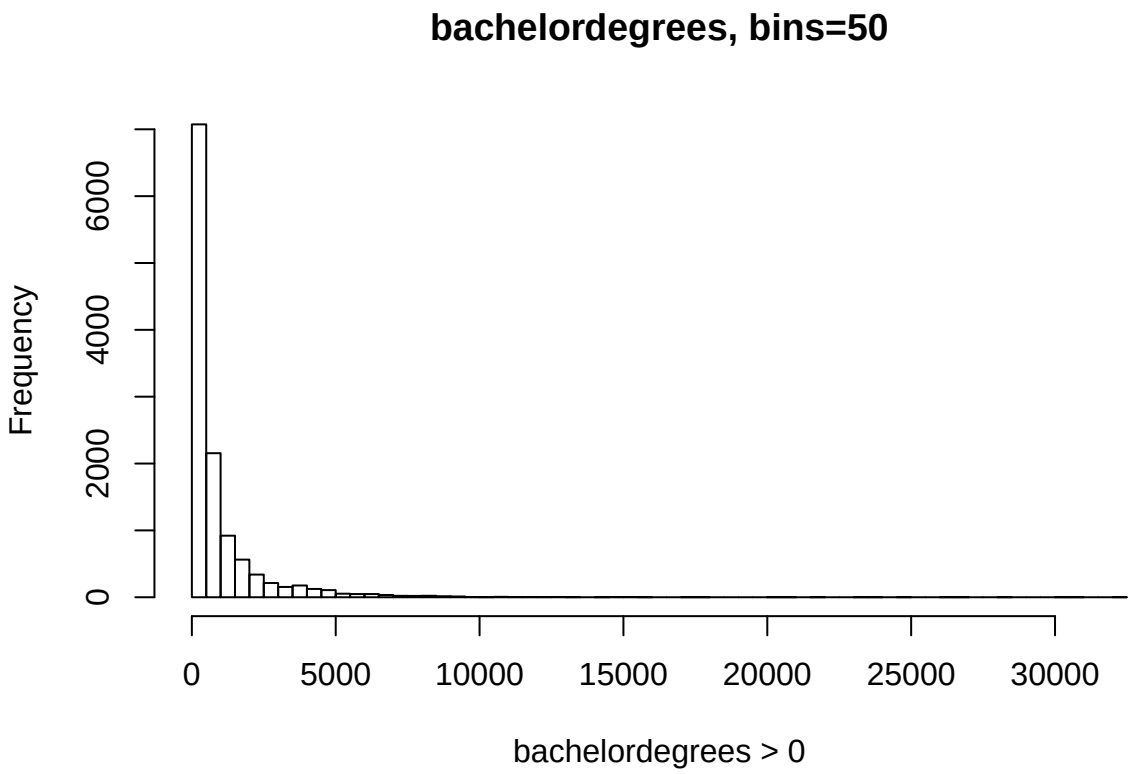

```r
length(data_no_NA$bachelordegrees[data_no_NA$bachelordegrees == 0])
```

```
## [1] 2615
```

```r
length(data_no_NA$bachelordegrees[data_no_NA$bachelordegrees > 10000])
```

```
## [1] 45
```

```r
hist(data_no_NA$bachelordegrees[data_no_NA$bachelordegrees > 0],
     main='bachelordegrees, bins=50', xlab='bachelordegrees > 0', breaks=50)
```



A sizable minority of rows (2615) belong to institutions that did not grant any bachelor's degrees. We suspect that these belong to smaller institutions that do no grant bachelor's degrees. Since we want to focus only on institutions that award bachelor's degrees and higher, these rows will be omitted.

While there are some observations where `bachelordegrees` is extremely high (greater than 10,000), a list of largest universities by undergraduate enrollment shows that these are plausible values [6].

3.3.5 Cleaning `grad_rate_150_p` and `grad_rate_150_n`

A close examination of `grad_rate_150_p` shows that, as a percentage, most values are between 0 and 1. A small minority of 10 observations are greater than 1, however.

```
summary(data_no_NA$grad_rate_150_p)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  0.00637  0.35370  0.50000  0.51040  0.64720 86.09000
```

```
describe(data_no_NA$grad_rate_150_p)
```

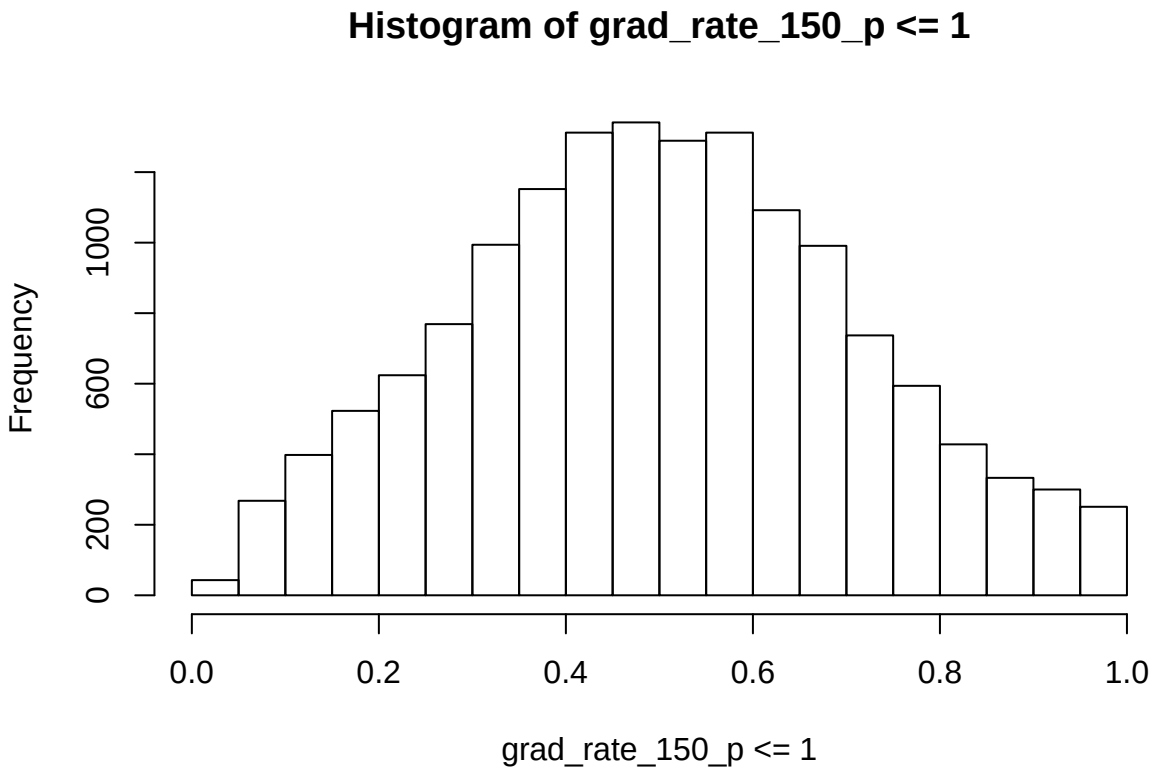
```
## data_no_NA$grad_rate_150_p
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##   14761       0   11280       1  0.5104  0.1538  0.2227  0.3537  0.5000
##     .75     .90     .95
##  0.6472  0.7874  0.8717
##
```

```
## lowest :  0.006369  0.009615  0.012821  0.013158  0.016245
## highest:  3.778145  3.894802  6.477437  7.500719 86.087456
```

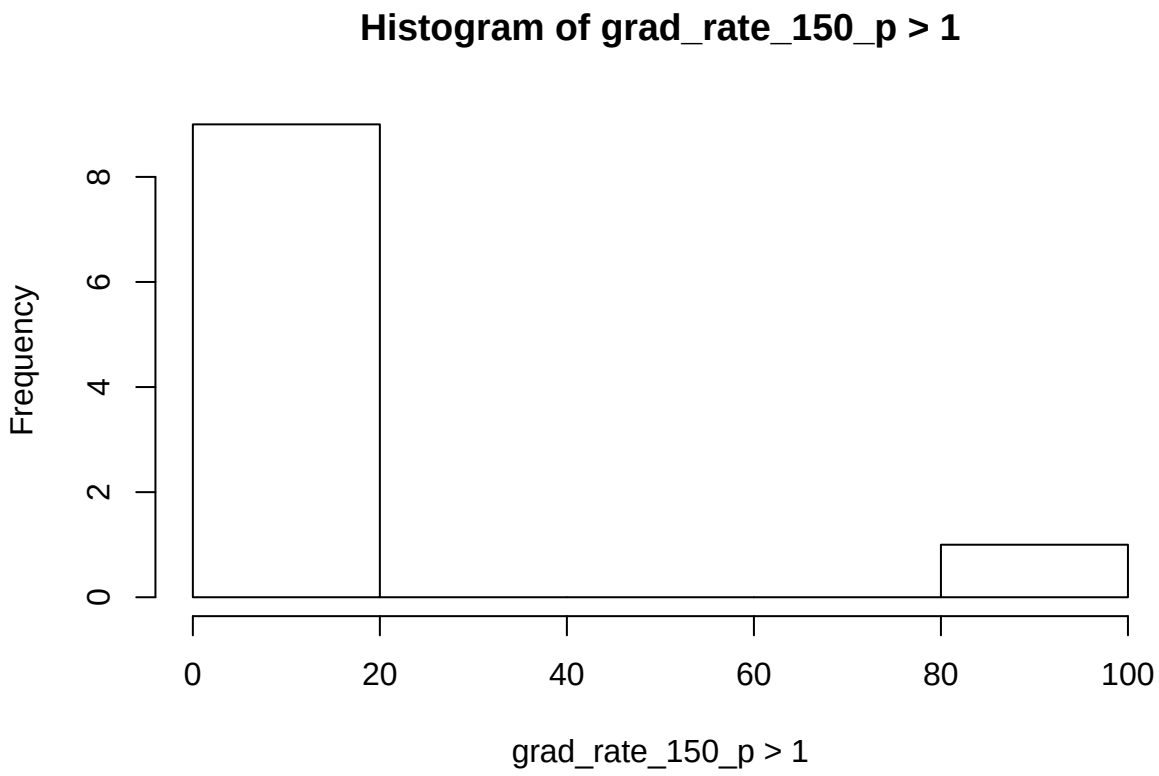
```
length(data_no_NA$grad_rate_150_p[data_no_NA$grad_rate_150_p>1])
```

```
## [1] 10
```

```
hist(data_no_NA$grad_rate_150_p[data_no_NA$grad_rate_150_p <= 1],
     xlab="grad_rate_150_p <= 1",
     main="Histogram of grad_rate_150_p <= 1")
```




```
hist(data_no_NA$grad_rate_150_p[data_no_NA$grad_rate_150_p > 1],
     xlab="grad_rate_150_p > 1", main="Histogram of grad_rate_150_p > 1")
```




```
#data_no_NA[data_no_NA$grad_rate_150_p > 1, ]
```

The rows where `grad_rate_150_p` is greater than 1 also have values of `grad_rate_150_n` that are not whole numbers. Since `grad_rate_150_n` is a count of students that graduated, we would expect this to be whole numbers. Overall, there are 48 rows where `grad_rate_150_n` is not a whole number, which include all 10 rows where `grad_rate_150_p` is greater than 1.

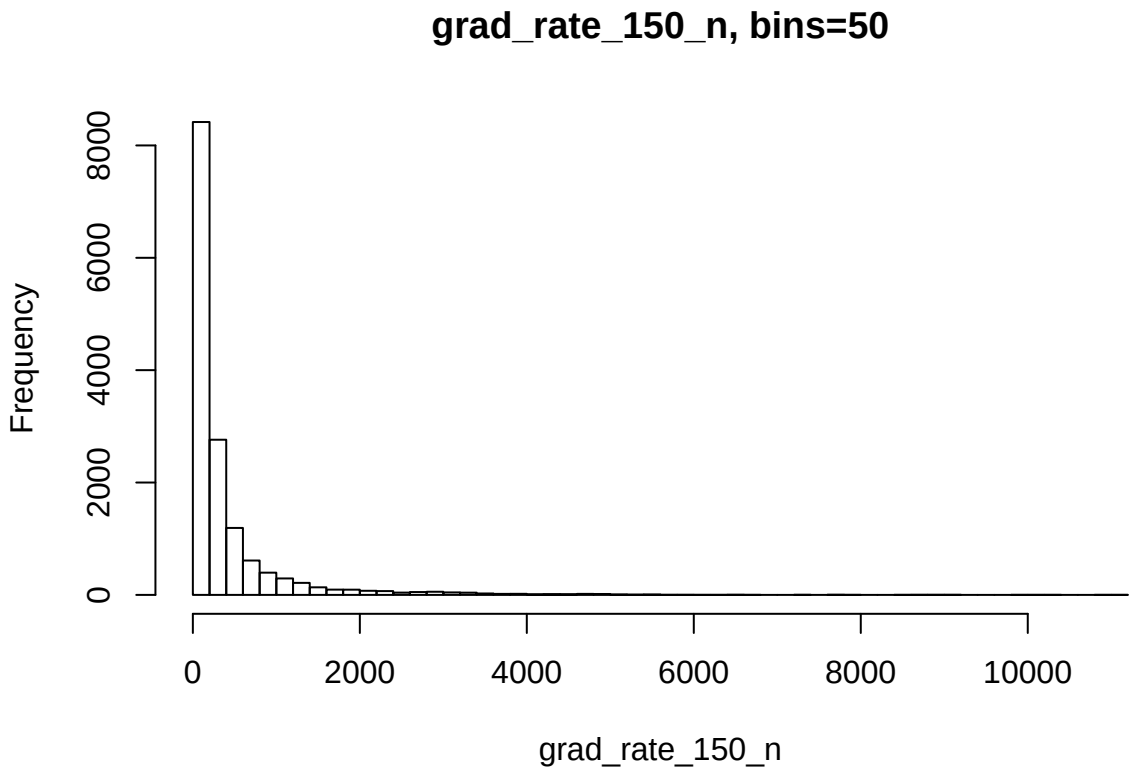
```
summary(data_no_NA$grad_rate_150_n)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     1.0    63.0   159.0   393.8   388.0 11190.0
```

```
describe(data_no_NA$grad_rate_150_n)
```

```
## data_no_NA$grad_rate_150_n
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##   14761       0    2040       1   393.8       9      19      63     159
##     .75     .90     .95
##     388     947    1604
##
## lowest :     1.000     1.097     1.218     1.355     1.547
## highest:  9895.000 10056.000 10391.000 10993.000 11194.000
```

```r
hist(data_no_NA$grad_rate_150_n,
     xlab="grad_rate_150_n", main="grad_rate_150_n, bins=50", breaks=50)
```




```r
length(data_no_NA$grad_rate_150_n[data_no_NA$grad_rate_150_n%%1 != 0])
```

```
## [1] 48
```

```r
data_no_NA$grad_rate_150_n[data_no_NA$grad_rate_150_n%%1 != 0]
```

```
##  [1]  59.658428   1.547003   1.354728   6.186635   6.958271   3.846511
##  [7]   2.193187  34.658646   3.996447   4.317727  41.231411  10.808457
## [13]   3.223120   7.556290   3.894802   2.131068  16.526005   1.096641
## [19]  16.510792  14.067477   2.665530  16.305826   1.218222  11.669627
## [25]   7.487404   3.005619   1.354728   2.159807   6.589529   9.961289
## [31] 842.826904  22.940487  11.111615  27.917206 358.177490   3.043353
## [37]  12.954875   1.637247   2.520972   6.772627  15.001438  44.133327
## [43]   1.856921   2.349813  86.087456  16.115417   5.441679   2.753698
```

```r
data_no_NA[data_no_NA$grad_rate_150_p > 1, c("X")]
```

```
##  [1] 10591 10942 12853 18611 22129 24016 54185 65576 75522 82780
```

```
data_no_NA[data_no_NA$grad_rate_150_n%%1 != 0, c("X")]
```

```
##  [1]  6639  6893  6917  7641  8094  8809  8811 10591 10942 12803 12853
## [12] 17432 18374 18611 22129 24016 25150 29870 36817 36820 39390 41817
## [23] 50253 51966 52939 54066 54185 57290 57292 57744 59230 59269 59442
## [34] 59461 59571 64914 65576 69688 73434 73436 75522 75672 76025 78470
## [45] 82780 83506 83929 85582
```

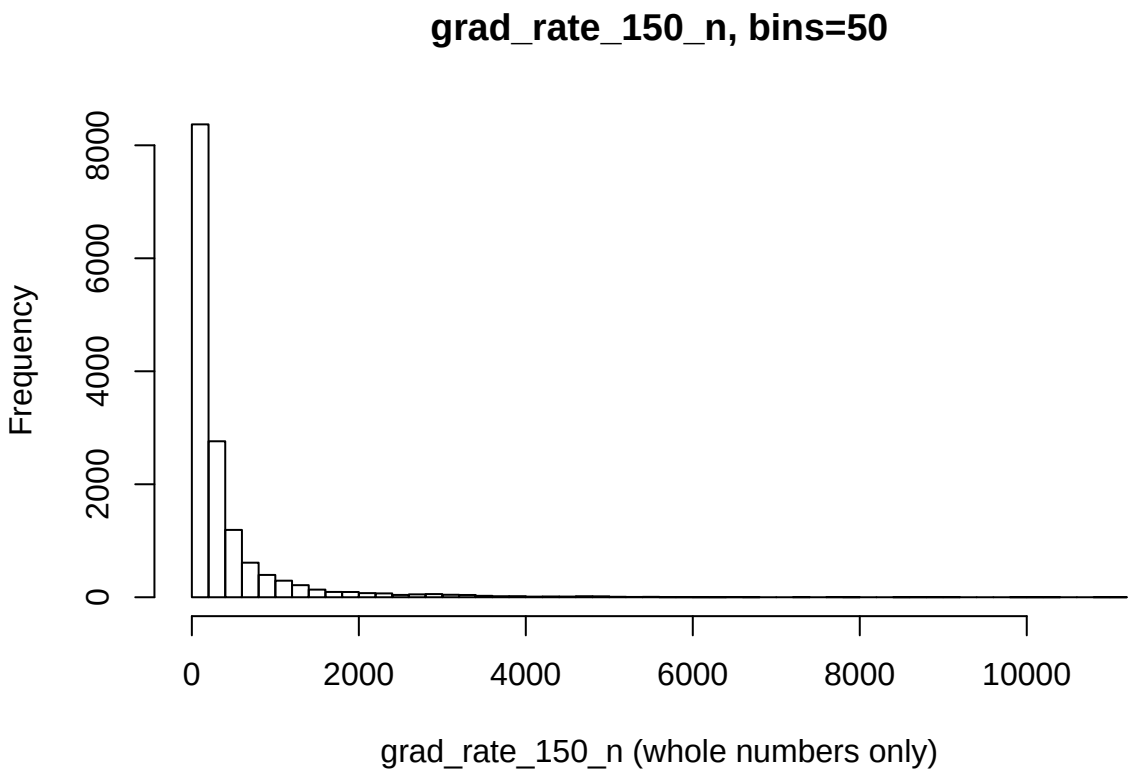
```
describe(data_no_NA$grad_rate_150_n[data_no_NA$grad_rate_150_n%%1 != 0])
```

```
## data_no_NA$grad_rate_150_n[data_no_NA$grad_rate_150_n%%1 != 0]
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##      48       0      47       1   36.83   1.355   1.610   2.629   6.681
##     .75     .90     .95
##  16.163  42.102  76.837
##
## lowest :   1.097   1.218   1.355   1.547   1.637
## highest:  44.133  59.658  86.087 358.177 842.827
```

```
describe(data_no_NA$grad_rate_150_n[data_no_NA$grad_rate_150_n%%1 == 0])
```

```
## data_no_NA$grad_rate_150_n[data_no_NA$grad_rate_150_n%%1 == 0]
##       n missing  unique    Info    Mean     .05     .10     .25     .50
##   14713       0    1993       1   394.9       9      20      63     160
##     .75     .90     .95
##     390     949    1608
##
## lowest :     1     2     3     4     5
## highest:  9895 10056 10391 10993 11194
```

```
hist(data_no_NA$grad_rate_150_n[data_no_NA$grad_rate_150_n%%1 == 0],
     xlab="grad_rate_150_n (whole numbers only)", main="grad_rate_150_n, bins=50", breaks=50)
```



The rows where `grad_rate_150_n` is not a whole number have a smaller mean by a factor of 10.

We have no data that may explain or account for the suspicious values of `grad_rate_150_p` and `grad_rate_150_n`. As such, we decided to omit these rows, rather than try to transform or scale them.

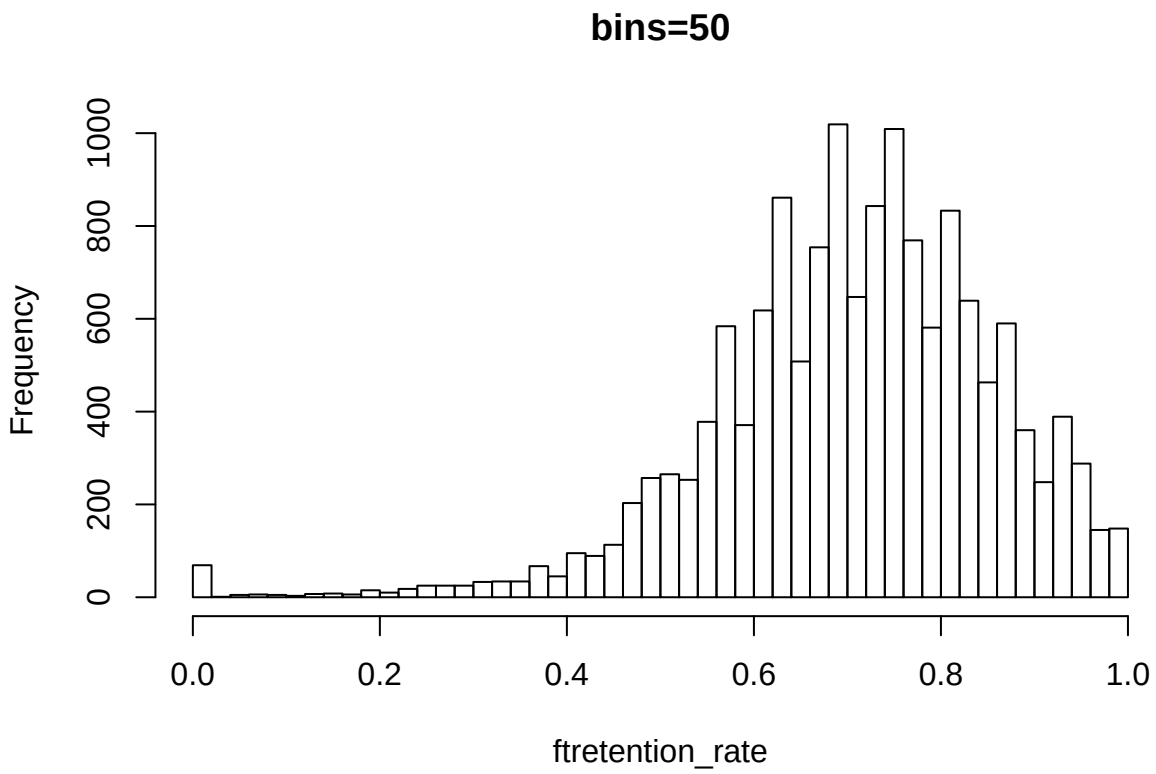
3.3.6 Cleaning `ftretention_rate`

A summary and histogram of `ftretention_rate` show no unusual values. It is a rate between 0 an 1.

```
summary(data_no_NA$ftretention_rate)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.6200  0.7200  0.7077  0.8100  1.0000
```

```
hist(data_no_NA$ftretention_rate, main='bins=50', xlab='ftretention_rate',
     breaks=50)
```



3.3.7 Removing Outliers

```
# Removing outliers and constraints percentages

data = data_no_NA[ (data_no_NA$academicyear == 2009) &
             (data_no_NA$fed_grant_num < 5000) & (0 < data_no_NA$fed_grant_num)  &
             (data_no_NA$fed_grant_pct <= 100) & (0 <= data_no_NA$fed_grant_pct) &
             (data_no_NA$fed_grant_avg_amount < 10000) &  (0 < data_no_NA$fed_grant_avg_amount)  &
             (data_no_NA$state_grant_num < 5000) & (0 < data_no_NA$state_grant_num)  &
             (data_no_NA$state_grant_pct <= 100) & (0 <= data_no_NA$state_grant_pct) &
             (data_no_NA$state_grant_avg_amount < 10000) & (0 < data_no_NA$state_grant_avg_amount ) &
             (data_no_NA$loan_num < 50000) & (0 < data_no_NA$loan_num) &
             (data_no_NA$loan_pct <= 100) & (0 <= data_no_NA$loan_pct) &
             (data_no_NA$loan_avg_amount < 30000) & (0 < data_no_NA$loan_avg_amount) &
             (0 < data_no_NA$bachelordegrees) &
             (data_no_NA$grad_rate_150_n%%1 == 0) &  (0 < data_no_NA$grad_rate_150_n) &
             (data_no_NA$grad_rate_150_p <= 1) ,  ]

# Unlike the other percentage variables, grad_rate_150_p is a proportion between 0 and 1.
# Whereas the other *_pct variables are percentages between 0 and 100.
# Multiply by 100 grad_rate_150_p
# Then a 1 unit change means a 1 percentage point change, which is probably
# much more meaningful and useful.
```

```r
data$grad_rate_150_p <- data$grad_rate_150_p * 100

str(data)
```

```
## 'data.frame':    1417 obs. of  15 variables:
##  $ X                     : int  2026 2725 2922 3321 3644 3919 4029 5116 5378 6156 ...
##  $ academicyear          : int  2009 2009 2009 2009 2009 2009 2009 2009 2009 2009 ...
##  $ fed_grant_num         : int  15 458 38 95 101 18 55 33 32 1 ...
##  $ fed_grant_pct         : int  10 94 60 16 64 56 24 57 23 1 ...
##  $ fed_grant_avg_amount  : int  2702 5580 3897 4049 3527 2427 3837 4185 3543 1731 ...
##  $ state_grant_num       : int  5 1 58 15 2 2 29 5 15 10 ...
##  $ state_grant_pct       : int  3 0 92 3 1 6 13 9 11 13 ...
##  $ state_grant_avg_amount: int  1599 500 1133 7067 7938 2250 2796 1683 3150 1500 ...
##  $ loan_num              : int  17 479 56 246 129 5 105 41 103 10 ...
##  $ loan_pct              : int  12 99 89 42 82 16 47 71 75 13 ...
##  $ loan_avg_amount       : int  14477 9645 8624 12031 8210 3692 7750 9041 14663 4228 ...
##  $ bachelordegrees       : num  40 25 320 314 22 26 64 50 139 62 ...
##  $ grad_rate_150_n       : num  230 7 27 196 32 8 29 2 89 68 ...
##  $ grad_rate_150_p       : num  62.5 43.8 50 79 45.1 ...
##  $ ftretention_rate      : num  0.88 0.36 0.59 0.7 0.59 0.3 0.6 0.29 0.56 1 ...
```

```r
summary(data)
```

```
##        X          academicyear  fed_grant_num    fed_grant_pct
##  Min.   : 2026   Min.   :2009   Min.   :   1.0   Min.   :  1.00
##  1st Qu.:37454   1st Qu.:2009   1st Qu.:  68.0   1st Qu.: 20.00
##  Median :53640   Median :2009   Median : 124.0   Median : 31.00
##  Mean   :52801   Mean   :2009   Mean   : 251.9   Mean   : 35.08
##  3rd Qu.:67694   3rd Qu.:2009   3rd Qu.: 297.0   3rd Qu.: 44.00
##  Max.   :87032   Max.   :2009   Max.   :4799.0   Max.   :100.00
##  fed_grant_avg_amount state_grant_num  state_grant_pct
##  Min.   : 485         Min.   :   1.0   Min.   :  0.00
##  1st Qu.:3893         1st Qu.:  50.0   1st Qu.: 18.00
##  Median :4285         Median : 131.0   Median : 31.00
##  Mean   :4341         Mean   : 295.5   Mean   : 35.62
##  3rd Qu.:4771         3rd Qu.: 323.0   3rd Qu.: 51.00
##  Max.   :9926         Max.   :4819.0   Max.   :100.00
##  state_grant_avg_amount    loan_num          loan_pct      loan_avg_amount
##  Min.   : 150           Min.   :    2.0   Min.   :  1.00   Min.   :  712
##  1st Qu.:2022           1st Qu.:  140.0   1st Qu.: 50.00   1st Qu.: 5466
##  Median :3018           Median :  279.0   Median : 66.00   Median : 6642
##  Mean   :3258           Mean   :  475.3   Mean   : 63.22   Mean   : 6956
##  3rd Qu.:4221           3rd Qu.:  550.0   3rd Qu.: 78.00   3rd Qu.: 8025
##  Max.   :9750           Max.   :10078.0   Max.   :100.00   Max.   :20824
##  bachelordegrees   grad_rate_150_n  grad_rate_150_p   ftretention_rate
##  Min.   :    2.0   Min.   :   1.0   Min.   :  2.985   Min.   :0.0000
##  1st Qu.:  183.0   1st Qu.:  70.0   1st Qu.: 40.359   1st Qu.:0.6400
##  Median :  399.0   Median : 184.0   Median : 52.830   Median :0.7300
##  Mean   :  875.4   Mean   : 443.5   Mean   : 53.862   Mean   :0.7229
##  3rd Qu.:  955.0   3rd Qu.: 466.0   3rd Qu.: 66.478   3rd Qu.:0.8200
##  Max.   :12723.0   Max.   :8787.0   Max.   :100.000   Max.   :1.0000
```

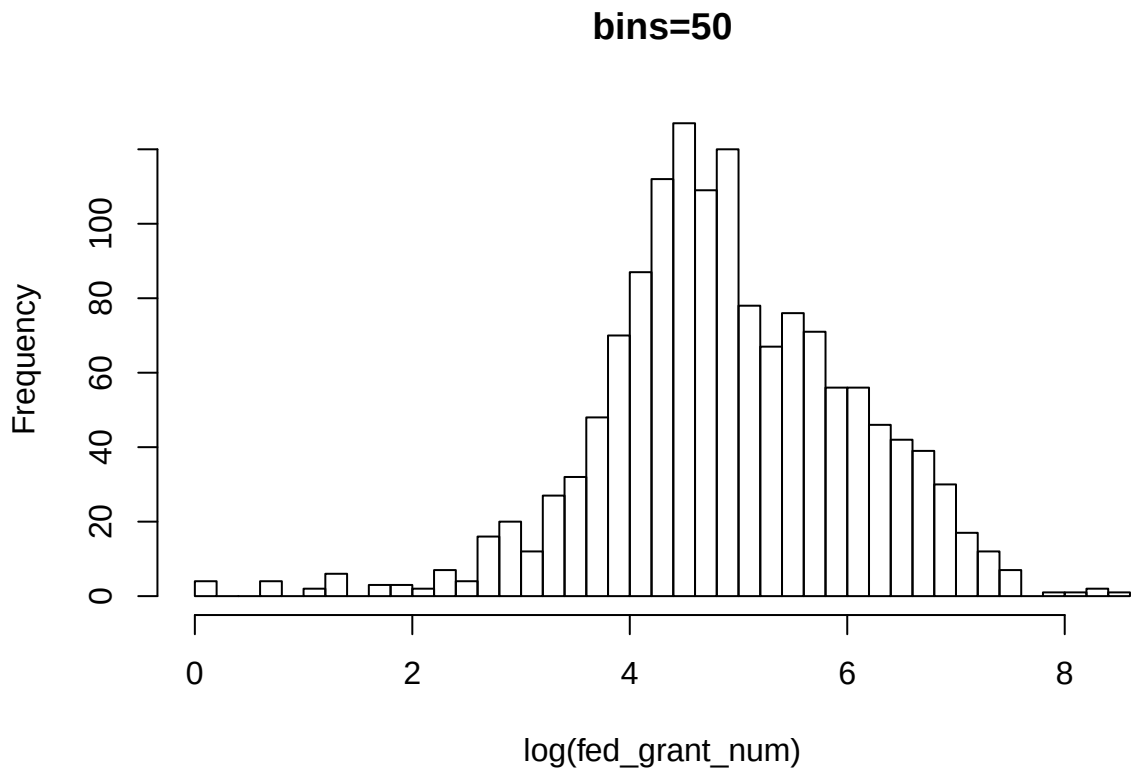
How many rows remain after removing outliers?

```r
nrow(data)
```

```
## [1] 1417
```

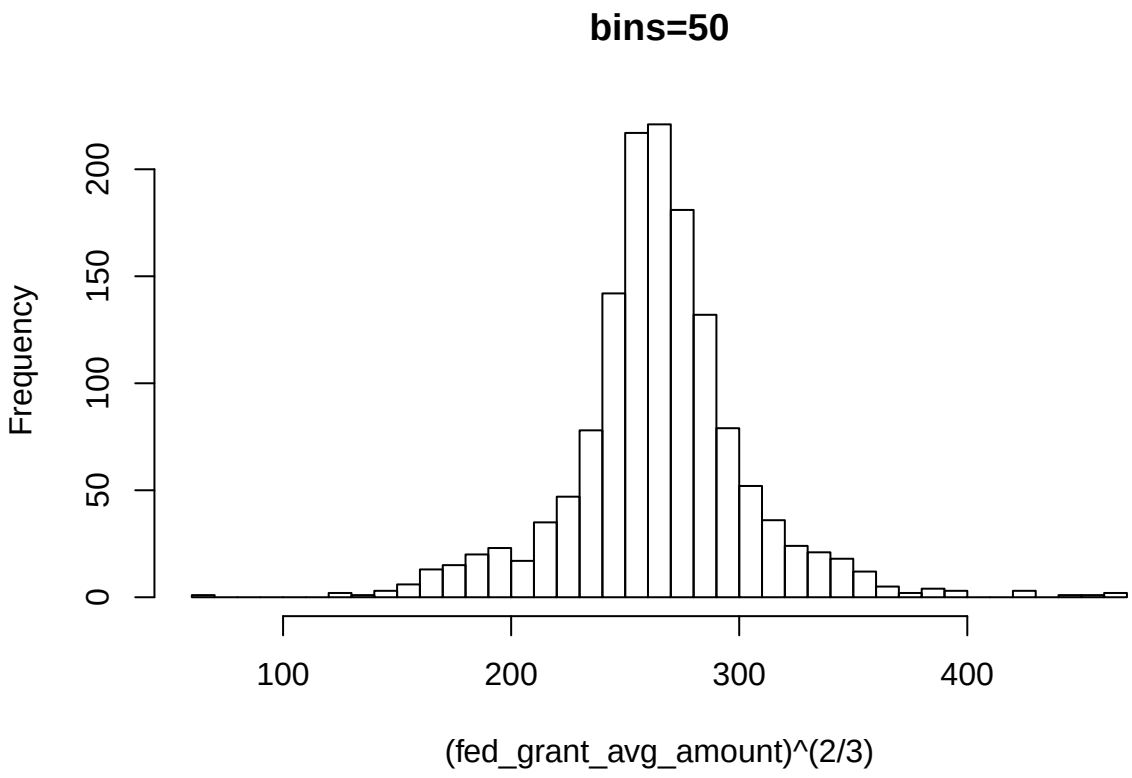
Perform transformation to Normal Distribution

```r
data_TF = data
hist(log(data$fed_grant_num), main='bins=50', xlab='log(fed_grant_num)',
breaks=50)
```



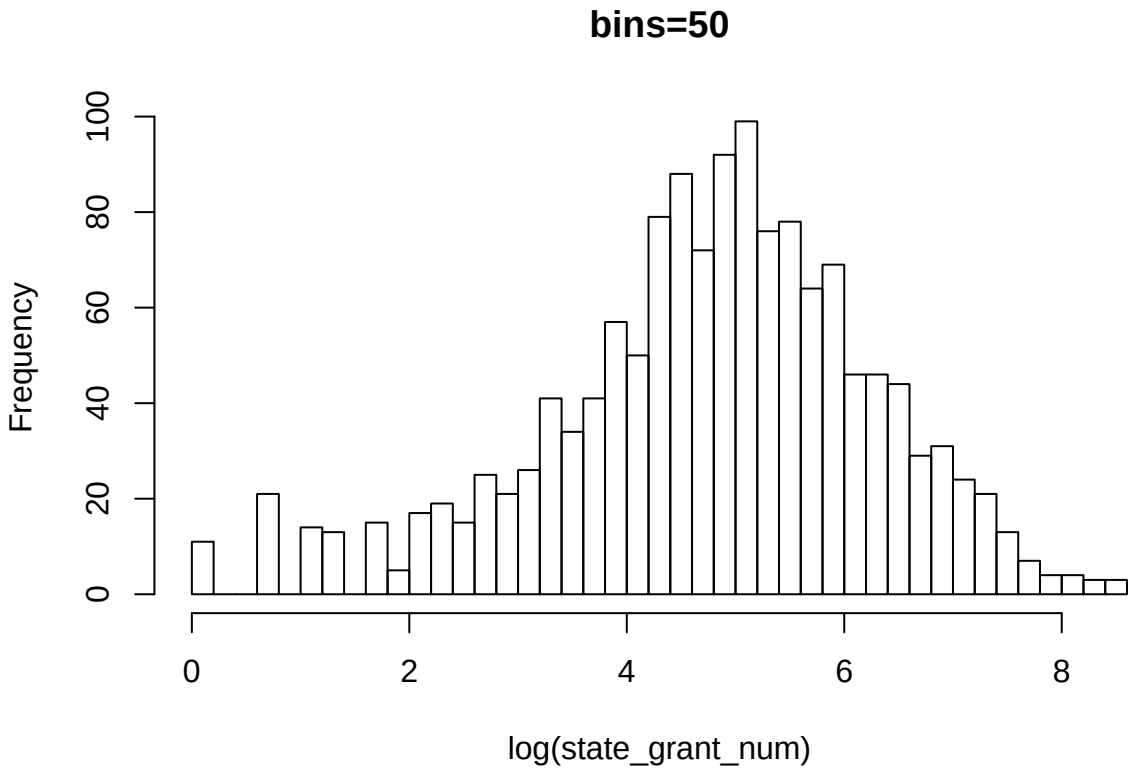

```r
data_TF$fed_grant_num = log(data$fed_grant_num)

hist((data$fed_grant_avg_amount)^(2/3), main='bins=50', xlab='(fed_grant_avg_amount)^(2/3)',
breaks=50)
```



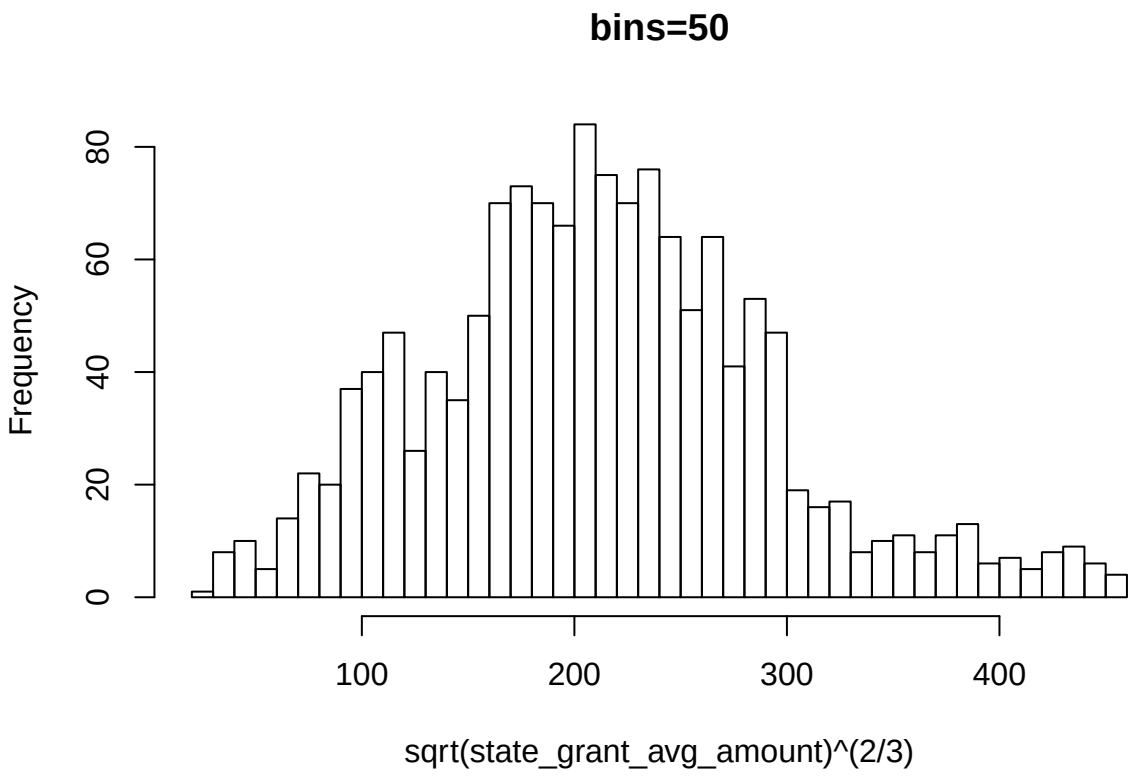

```
data_TF$fed_grant_avg_amount = (data$fed_grant_avg_amount)^(2/3)

hist(log(data$state_grant_num), main='bins=50', xlab='log(state_grant_num)',
breaks=50)
```



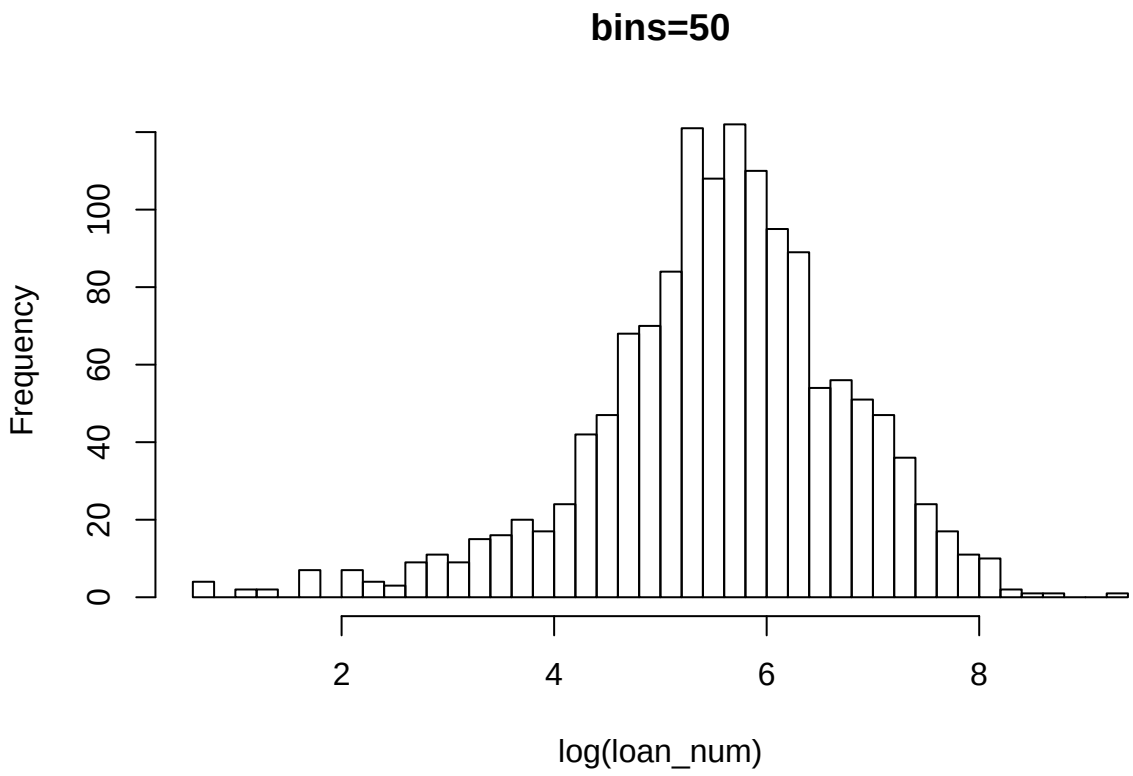

```
data_TF$state_grant_num = log(data$state_grant_num)

hist((data$state_grant_avg_amount)^(2/3), main='bins=50', xlab='sqrt(state_grant_avg_amount)^(2/3)',
breaks=50)
```



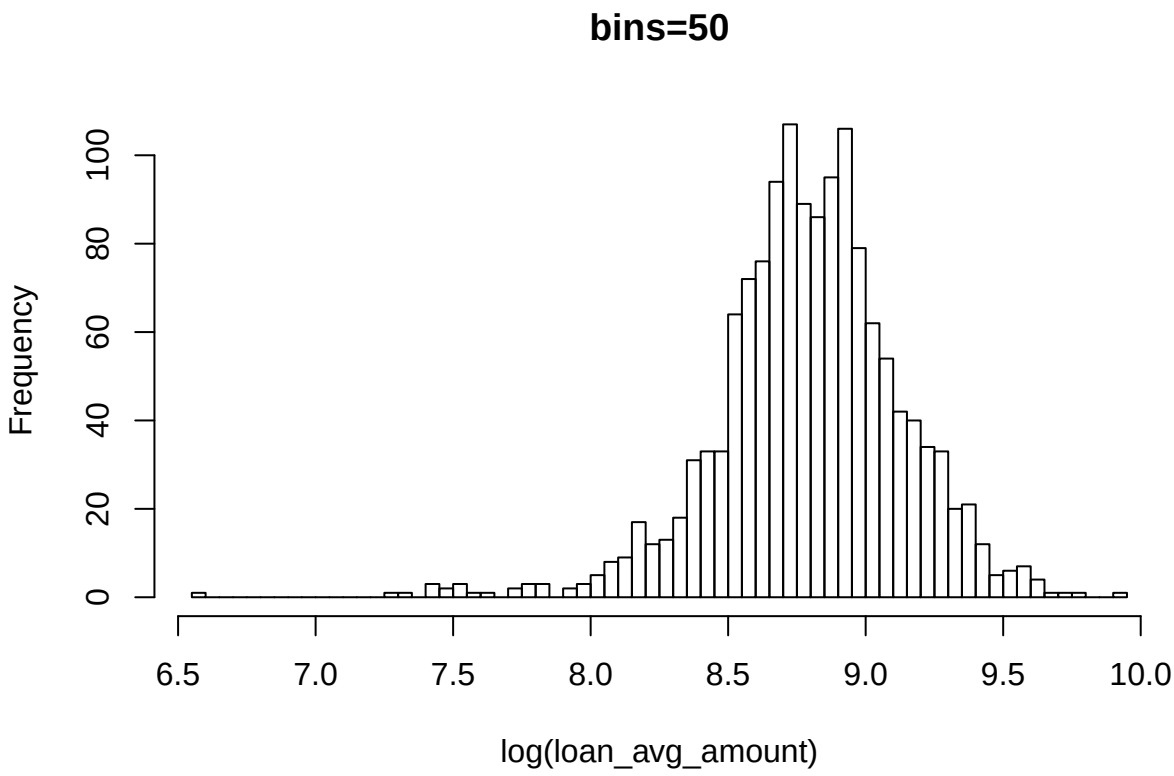

```
data_TF$state_grant_avg_amount = log(data$state_grant_avg_amount)

hist(log(data$loan_num), main='bins=50', xlab='log(loan_num)',
breaks=50)
```



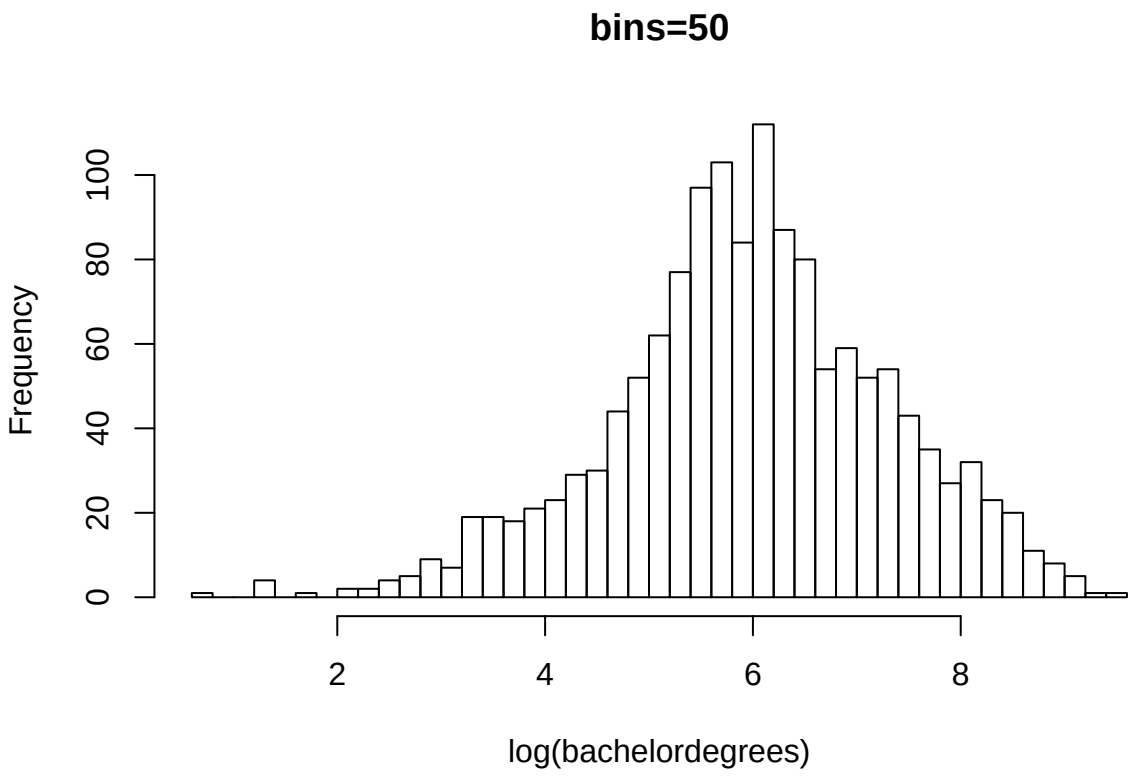

```
data_TF$loan_num = log(data$loan_num)

hist(log(data$loan_avg_amount), main='bins=50', xlab='log(loan_avg_amount)',
breaks=50)
```



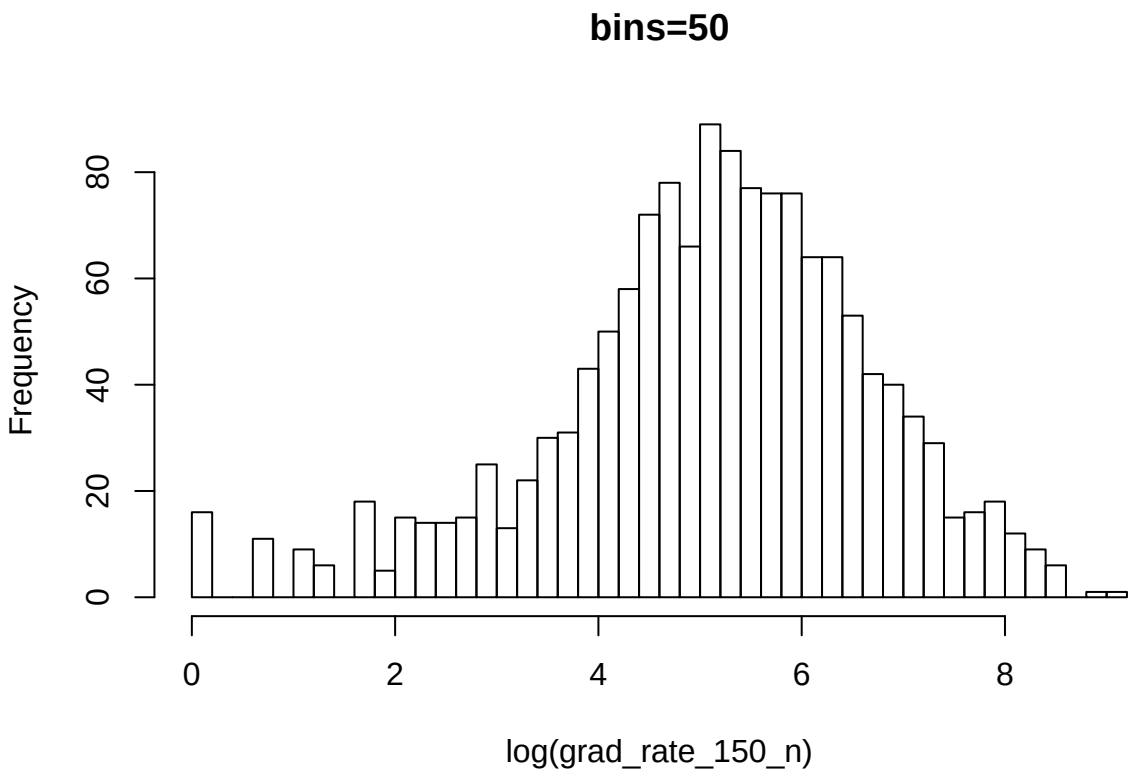

```
data_TF$loan_avg_amount = log(data$loan_avg_amount)

hist(log(data$bachelordegrees), main='bins=50', xlab='log(bachelordegrees)',
breaks=50)
```




```
data_TF$bachelordegrees = log(data$bachelordegrees)

hist(log(data$grad_rate_150_n), main='bins=50', xlab='log(grad_rate_150_n)',
breaks=50)
```




```
data_TF$grad_rate_150_n = log(data$grad_rate_150_n)
```

3.4 Modeling

We have three variables related to graduation rate: `bachelordegrees`, `grad_rate_150_n`, and `grad_rate_150_p`. It is important to note that `bachelordegrees` and `grad_rate_150_n` are absolute counts, whereas `grad_rate_150_p` is a percentage between 0 and 100.

We chose not to model `bachelordegrees` since our EDA shows it is very similar to `grad_rate_150_n` and `grad_rate_150_n` has a corresponding percentage variable.

We wish to assess whether any of these dependent variables can be predicted by the variables related to student aid. The independent variables at our disposal are: `fed_grant_num`, `fed_grant_pct`, `fed_grant_avg_amount`, `state_grant_num`, `state_grant_pct`, `state_grant_avg_amount`, `loan_num`, `loan_pct`, and `loan_avg_amount`. Here we have available absolute counts, percentages, and averages.

3.5 Using `grad_rate_150_n` as academic success measure

3.5.1 avg_amount independent vars

Start with most parsimonious model:

```
grad_num150_a_m1 <- lm(grad_rate_150_n ~ fed_grant_avg_amount, data = data_TF)
summary(grad_num150_a_m1)
```

```
##
## Call:
## lm(formula = grad_rate_150_n ~ fed_grant_avg_amount, data = data_TF)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.3332 -0.7979  0.0556  0.9225  3.8836
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          1.821936   0.274969   6.626 4.89e-11 ***
## fed_grant_avg_amount 0.012364   0.001028  12.031  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.519 on 1415 degrees of freedom
## Multiple R-squared:  0.0928, Adjusted R-squared:  0.09216
## F-statistic: 144.7 on 1 and 1415 DF,  p-value: < 2.2e-16
```

```
# statistically significant, positive coefficient, but low magnitude
#Adjusted R-squared:  0.09216

bptest(grad_num150_a_m1)
```

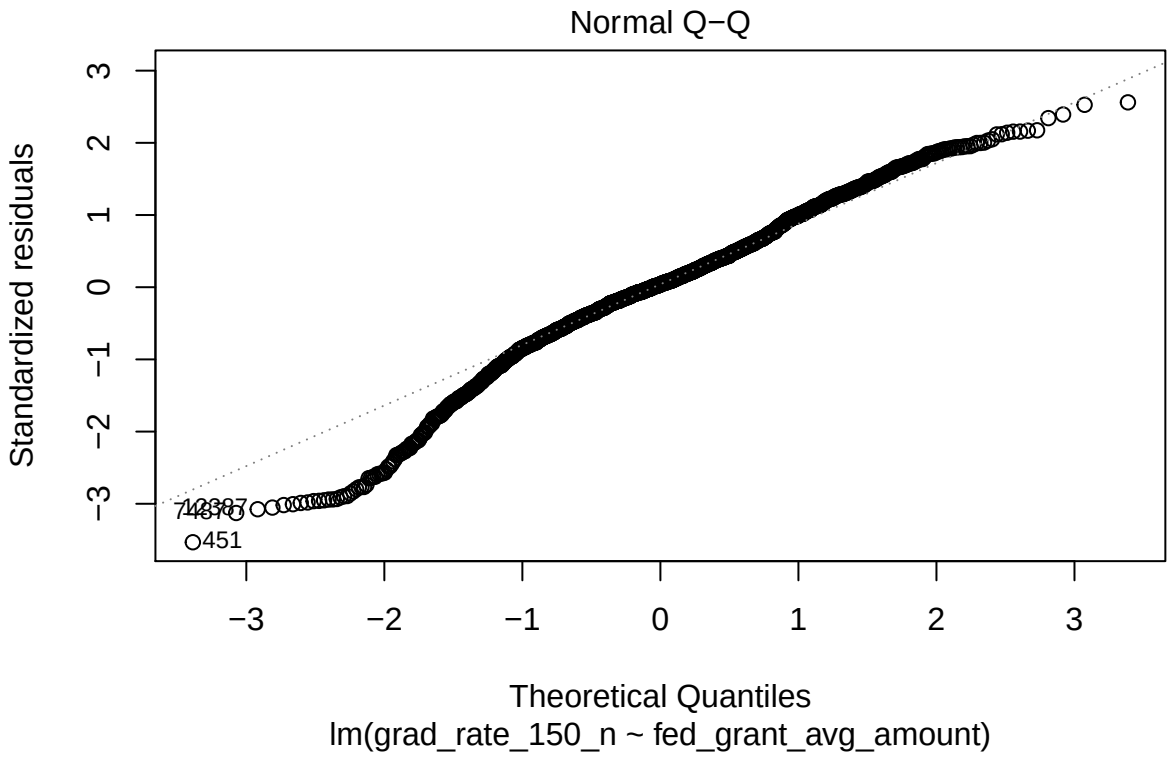
```
##
##  studentized Breusch-Pagan test
##
## data:  grad_num150_a_m1
## BP = 21.463, df = 1, p-value = 3.608e-06
```

```
# significant, so need heteroskedasticity-corrected standard errors to evaluate the model
coeftest(grad_num150_a_m1, vcov=vcovHC)
```

```
##
## t test of coefficients:
##
##                       Estimate Std. Error t value  Pr(>|t|)
## (Intercept)          1.8219360  0.2975973  6.1222 1.194e-09 ***
## fed_grant_avg_amount 0.0123637  0.0010857 11.3876 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

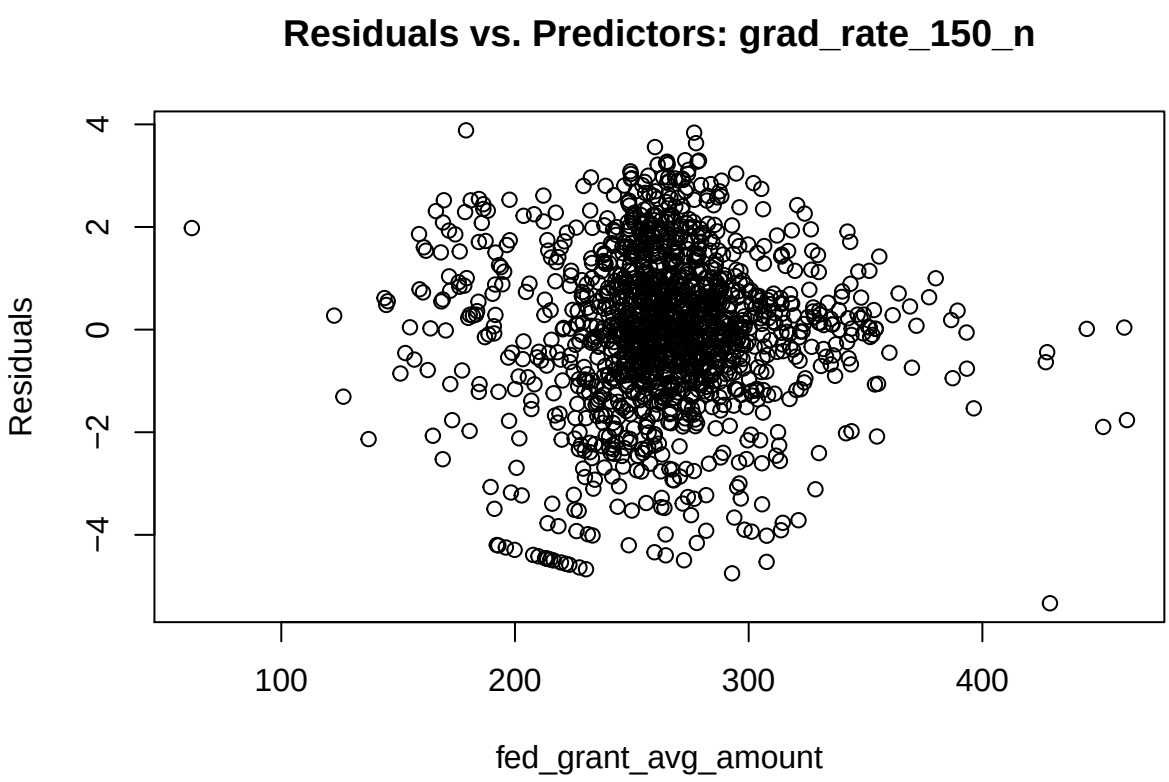
Assess the model diagnostics:

```
plot(grad_num150_a_m1, which=2)
```

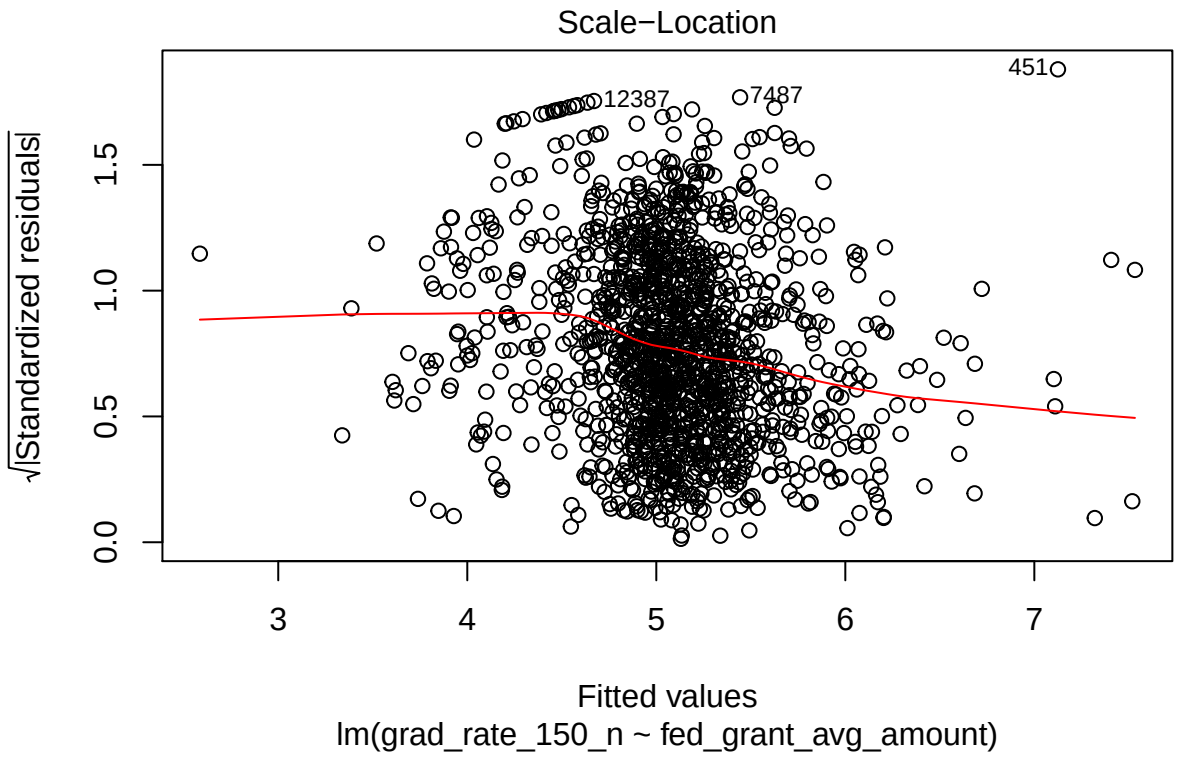



```
# Q-Q plot curves away from diagonal at left end
# Suggests non-normality of errors

# Residuals vs. Predictors plot
plot(data_TF$fed_grant_avg_amount, resid(grad_num150_a_m1),
     xlab="fed_grant_avg_amount", ylab="Residuals",
     main="Residuals vs. Predictors: grad_rate_150_n")
```

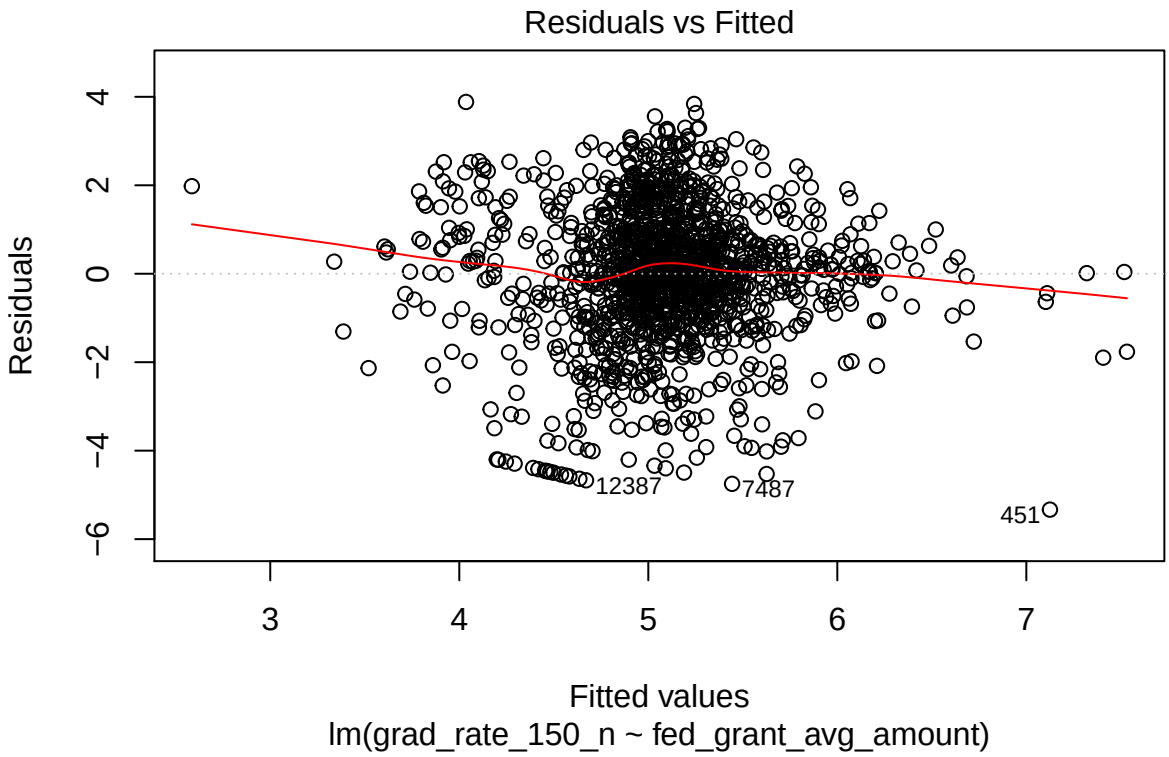



```
plot(grad_num150_a_m1, which=3)
```




```
# Scale-Location plot provides evidence of heteroskedasticity,
# since the red line bends downward.
# This is consistent with the results of bptest

plot(grad_num150_a_m1, which=1)
```




```
# Some non-flatness in red residual line and overall downward trend
```

Test whether `state_grant_avg_amount` improves the model:

```
grad_num150_a_m2 <- lm(grad_rate_150_n ~ fed_grant_avg_amount + state_grant_avg_amount, data = data_TF)
coeftest(grad_num150_a_m2, vcov=vcovHC)
```

```
##
## t test of coefficients:
##
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -0.2222113  0.5251785 -0.4231   0.6723
## fed_grant_avg_amount    0.0116741  0.0010931 10.6800  < 2e-16 ***
## state_grant_avg_amount  0.2815916  0.0618260  4.5546  5.7e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# significant model, both coefficients positive, but little practical effect
waldtest(grad_num150_a_m2, grad_num150_a_m1, vcov = vcovHC)
```

```
## Wald test
##
## Model 1: grad_rate_150_n ~ fed_grant_avg_amount + state_grant_avg_amount
## Model 2: grad_rate_150_n ~ fed_grant_avg_amount
```

```
##   Res.Df Df      F  Pr(>F)
## 1   1414
## 2   1415 -1 20.744 5.7e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We get a statistically significant F-statistic, so we reject the null hypothesis that `state_grant_avg_amount` has no effect. So `grad_num150_a_m2` is preferred over `grad_num150_a_m1`.

Test whether `loan_avg_amount` improves the model.

```
grad_num150_a_m3 <- lm(grad_rate_150_n ~ fed_grant_avg_amount + state_grant_avg_amount
                       + loan_avg_amount, data = data_TF)
coeftest(grad_num150_a_m3, vcov=vcovHC)
```

```
##
## t test of coefficients:
##
##                          Estimate Std. Error t value  Pr(>|t|)
## (Intercept)              5.927505   1.029405  5.7582 1.042e-08 ***
## fed_grant_avg_amount     0.012455   0.001085 11.4788 < 2.2e-16 ***
## state_grant_avg_amount   0.335025   0.062293  5.3782 8.794e-08 ***
## loan_avg_amount         -0.770948   0.124538 -6.1904 7.847e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# significant model, but small magnitude
# loan_avg_amount has negative coefficient
waldtest(grad_num150_a_m3, grad_num150_a_m2, vcov = vcovHC)
```

```
## Wald test
##
## Model 1: grad_rate_150_n ~ fed_grant_avg_amount + state_grant_avg_amount +
##     loan_avg_amount
## Model 2: grad_rate_150_n ~ fed_grant_avg_amount + state_grant_avg_amount
##   Res.Df Df      F    Pr(>F)
## 1   1413
## 2   1414 -1 38.322 7.847e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

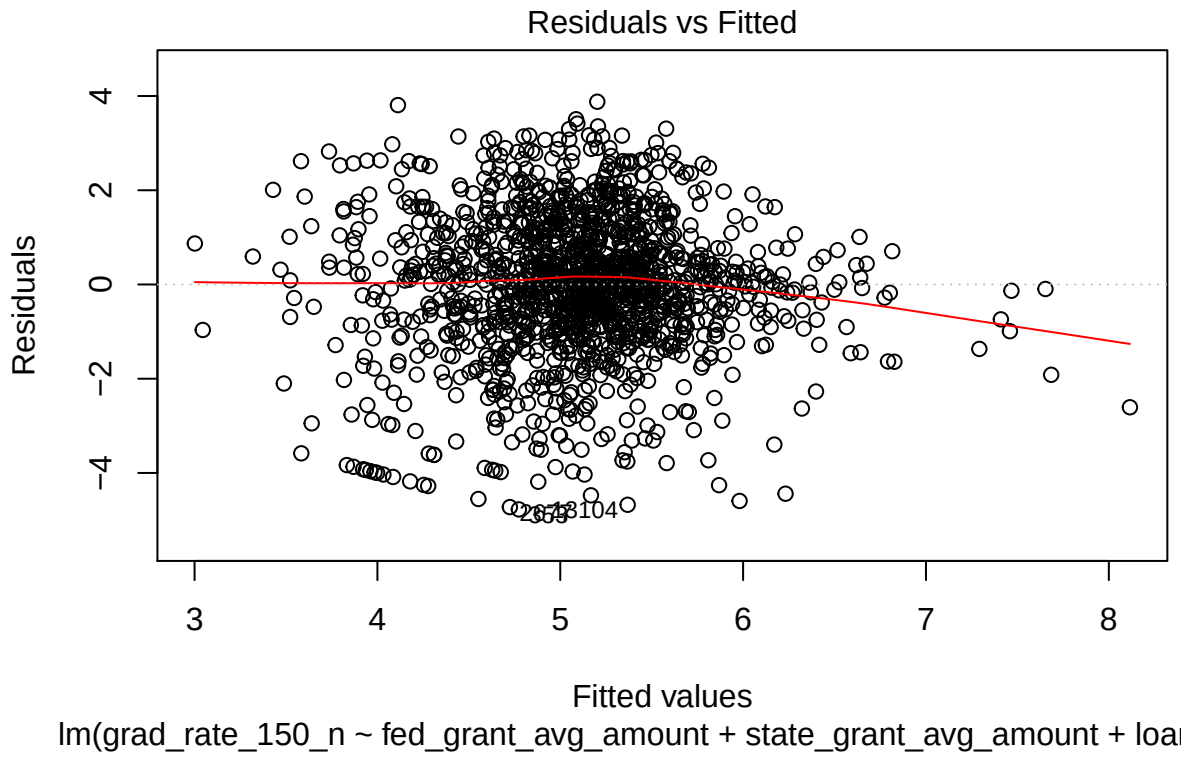
```
bptest(grad_num150_a_m3)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  grad_num150_a_m3
## BP = 32.761, df = 3, p-value = 3.617e-07
```

We get a statistically significant F-statistic, so we reject the null hypothesis that `loan_avg_amount` has no effect. So `grad_num150_a_m3` is preferred over `grad_num150_a_m2`.

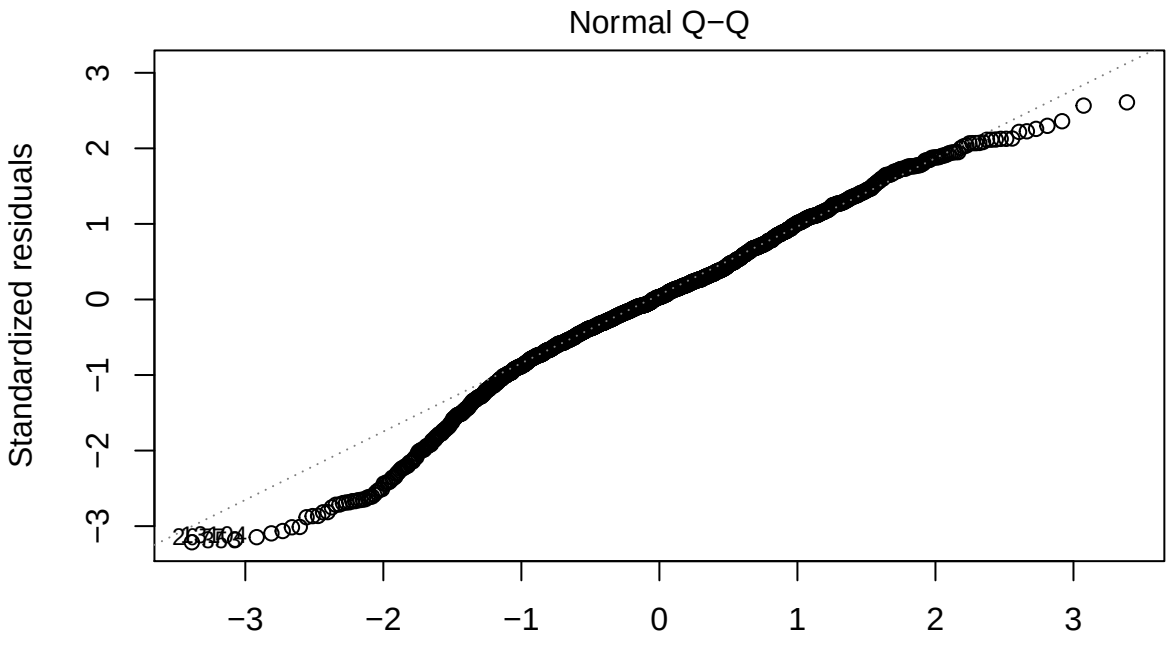
Assess model diagnostics:

```r
plot(grad_num150_a_m3, which=1)
```



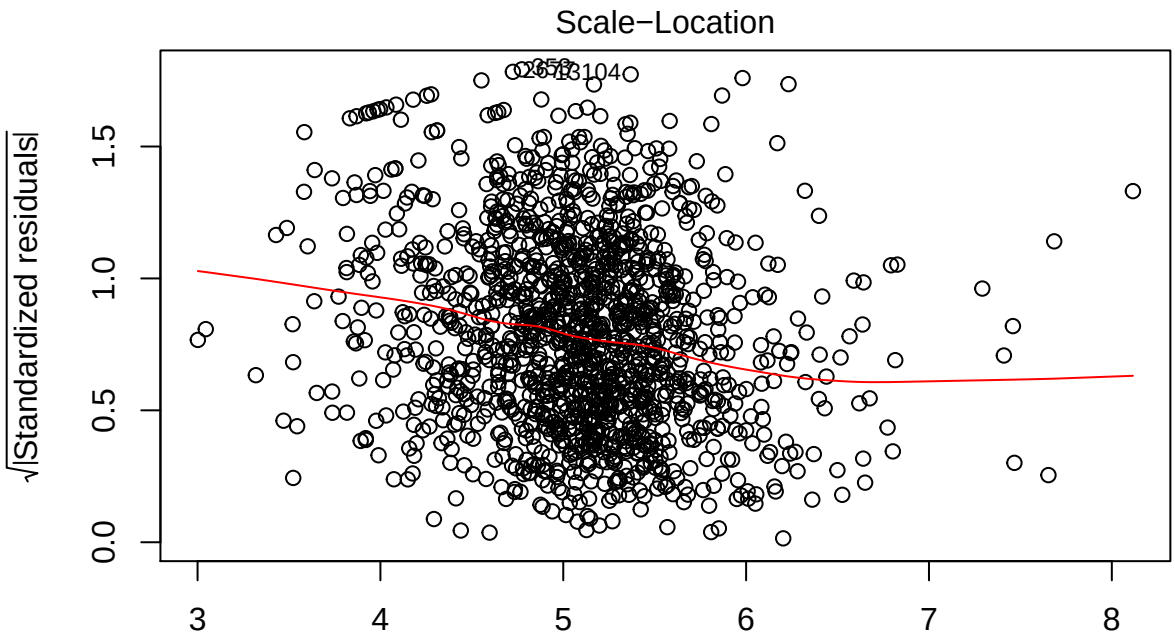

```r
# Red residual line more flat overall, bends down at right, fairly close to 0
# ZCM and linearity OK

plot(grad_num150_a_m3, which=2)
```




```
# Q-Q plot improved over grad_num150_a_m1 model

plot(grad_num150_a_m3, which=3)
```




```
# Scale-Location plot provides evidence of heteroskedasticity,
# since the red line non-flat at left end.
# This is consistent with the results of bptest
```

3.5.2 Percentage independent vars

```
grad_num150_p_m1 <- lm(grad_rate_150_n ~ fed_grant_pct, data = data_TF)
summary(grad_num150_p_m1)
```

```
##
## Call:
## lm(formula = grad_rate_150_n ~ fed_grant_pct, data = data_TF)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.0180 -0.6566  0.0264  0.8128  4.3960
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    6.631630   0.072572   91.38   <2e-16 ***
## fed_grant_pct -0.043830   0.001803  -24.31   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 1.34 on 1415 degrees of freedom
## Multiple R-squared:  0.2946, Adjusted R-squared:  0.2941
## F-statistic:   591 on 1 and 1415 DF,  p-value: < 2.2e-16
```

```
# statistically significant, negative coefficient
#Adjusted R-squared:  0.2941

bptest(grad_num150_p_m1)
```

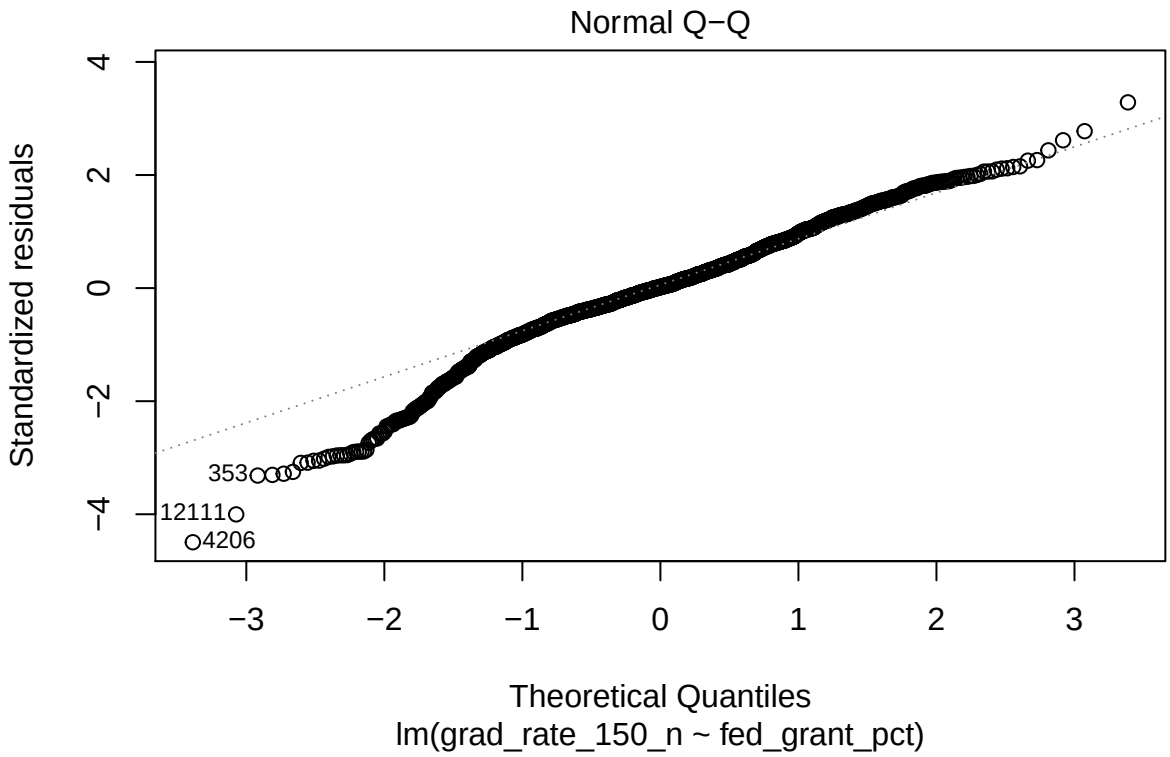
```
##
##  studentized Breusch-Pagan test
##
## data:  grad_num150_p_m1
## BP = 27.191, df = 1, p-value = 1.843e-07
```

```
coeftest(grad_num150_p_m1, vcov=vcovHC)
```

```
##
## t test of coefficients:
##
##                 Estimate Std. Error t value  Pr(>|t|)
## (Intercept)    6.6316300  0.0729601  90.894 < 2.2e-16 ***
## fed_grant_pct -0.0438302  0.0020497 -21.384 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

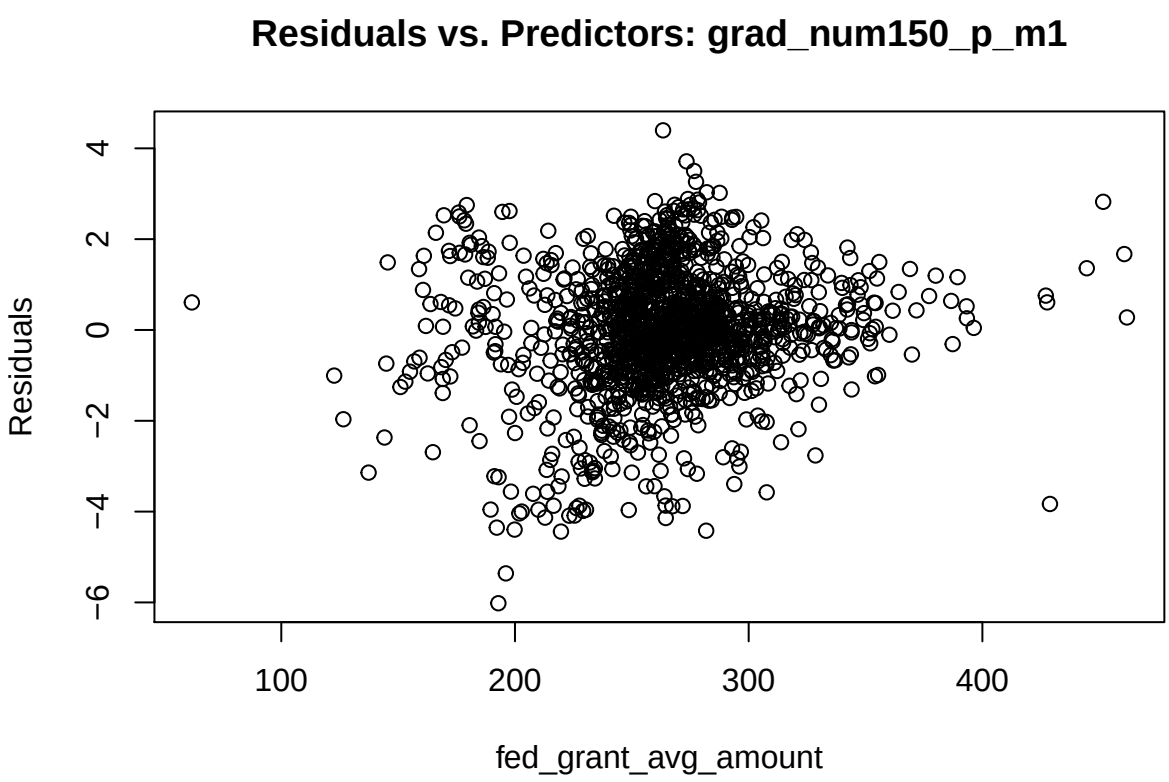
Assess the model diagnostics:

```
plot(grad_num150_p_m1, which=2)
```

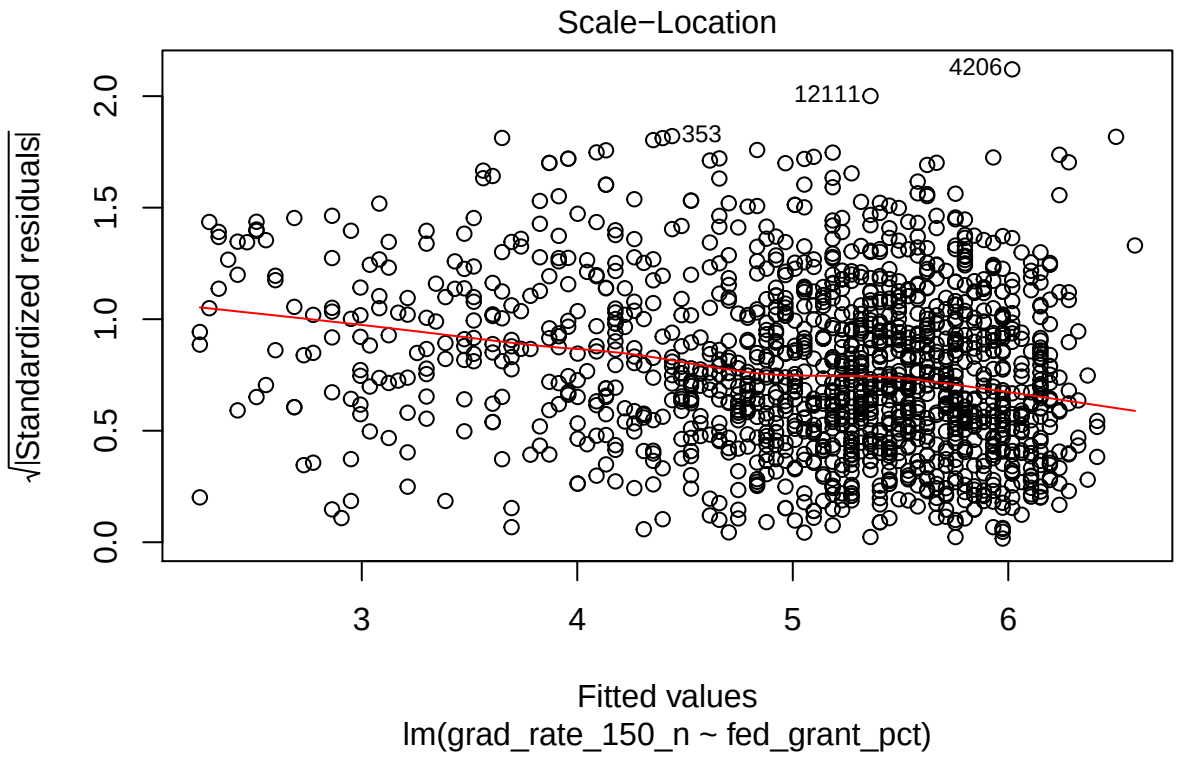



```
# Q-Q plot curves away from diagonal at left end
# Suggests non-normality of errors

# Residuals vs. Predictors plot
plot(data_TF$fed_grant_avg_amount, resid(grad_num150_p_m1),
     xlab="fed_grant_avg_amount", ylab="Residuals",
     main="Residuals vs. Predictors: grad_num150_p_m1")
```

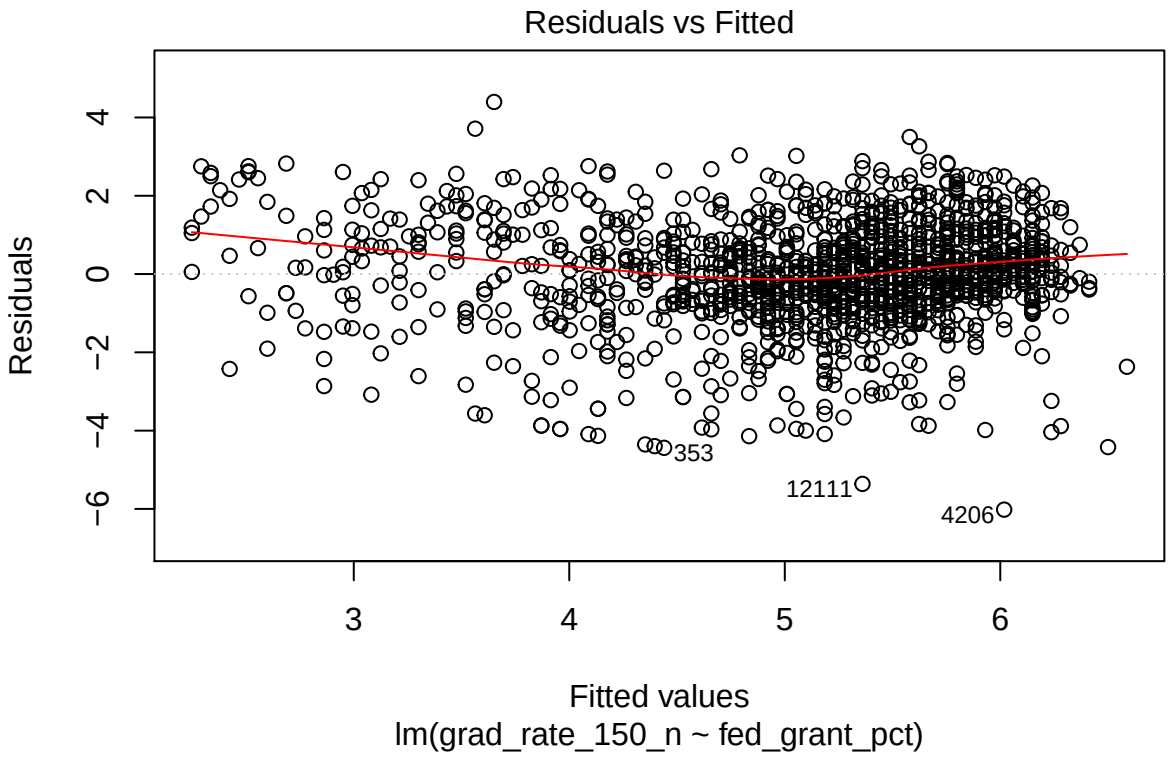



```
plot(grad_num150_p_m1, which=3)
```




```
# Scale-Location plot provides evidence of heteroskedasticity,
# since the red line bends downward.
# This is consistent with the results of bptest

plot(grad_num150_p_m1, which=1)
```




```
# Slight u-shape in red residual line, roughly even distribution in sign
# Overall fairly flat and close to 0. ZCM and linearity assumptions OK
```

Test whether `state_grant_pct` should be included:

```
grad_num150_p_m2 <- lm(grad_rate_150_n ~ fed_grant_pct + state_grant_pct, data = data_TF)
coeftest(grad_num150_p_m2, vcov=vcovHC)
```

```
##
## t test of coefficients:
##
##                   Estimate Std. Error  t value  Pr(>|t|)
## (Intercept)      6.4899488  0.0838137  77.4330 < 2.2e-16 ***
## fed_grant_pct   -0.0454206  0.0021289 -21.3356 < 2.2e-16 ***
## state_grant_pct  0.0055432  0.0016730   3.3132 0.0009455 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# significant model, fed negative, state positive, but little practical effect
waldtest(grad_num150_p_m2, grad_num150_p_m1, vcov = vcovHC)
```

```
## Wald test
##
## Model 1: grad_rate_150_n ~ fed_grant_pct + state_grant_pct
```

```
## Model 2: grad_rate_150_n ~ fed_grant_pct
##   Res.Df Df      F    Pr(>F)
## 1   1414
## 2   1415 -1 10.978 0.0009455 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# improves the model over m1
```

```
grad_num150_p_m3 <- lm(grad_rate_150_n ~ fed_grant_pct + state_grant_pct + loan_pct, data = data_TF)
coeftest(grad_num150_p_m3, vcov=vcovHC)
```

```
##
## t test of coefficients:
##
##                   Estimate Std. Error  t value  Pr(>|t|)
## (Intercept)      7.4318813  0.1330511  55.8574 < 2.2e-16 ***
## fed_grant_pct   -0.0398880  0.0019513 -20.4419 < 2.2e-16 ***
## state_grant_pct  0.0069743  0.0016130   4.3239  1.64e-05 ***
## loan_pct        -0.0187750  0.0019357  -9.6993 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# all 3 coefficients stat significant, fed_grant and loan negative, state grant positive
# all of little practical significance
waldtest(grad_num150_p_m3, grad_num150_p_m2, vcov = vcovHC)
```

```
## Wald test
##
## Model 1: grad_rate_150_n ~ fed_grant_pct + state_grant_pct + loan_pct
## Model 2: grad_rate_150_n ~ fed_grant_pct + state_grant_pct
##   Res.Df Df      F    Pr(>F)
## 1   1413
## 2   1414 -1 94.077 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

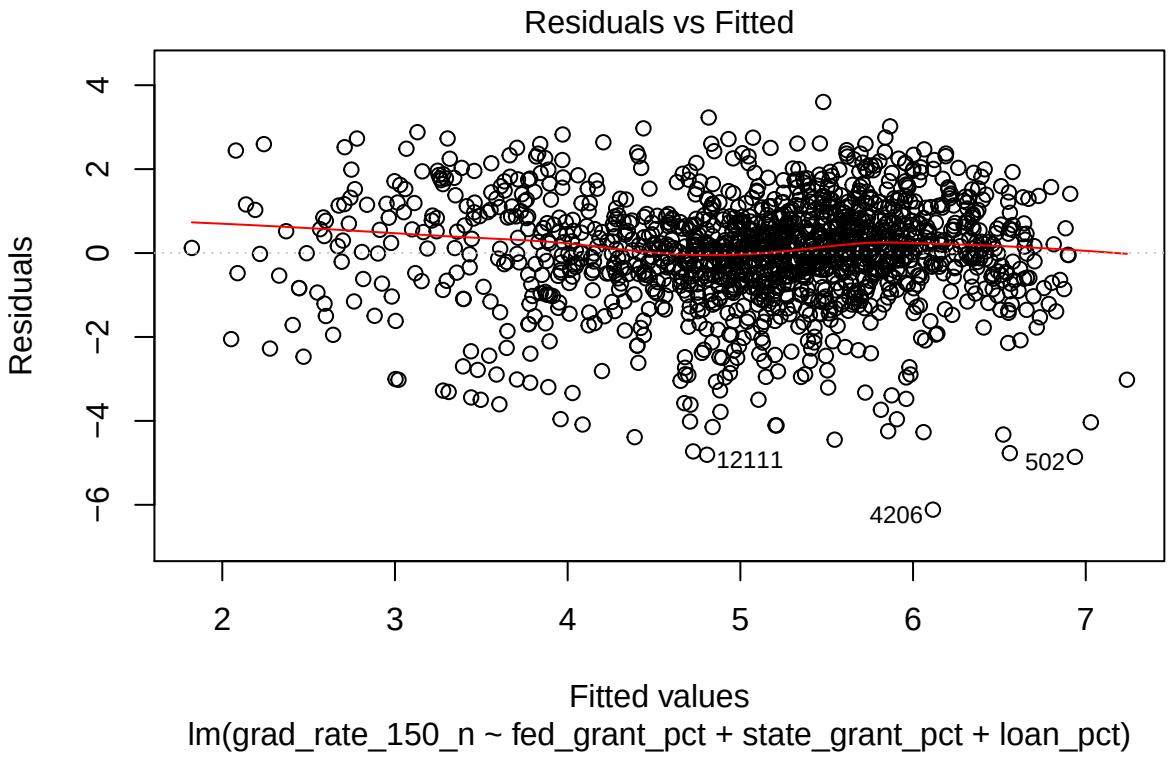
```
# improves model over m2

bptest(grad_num150_p_m3)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  grad_num150_p_m3
## BP = 27.815, df = 3, p-value = 3.972e-06
```

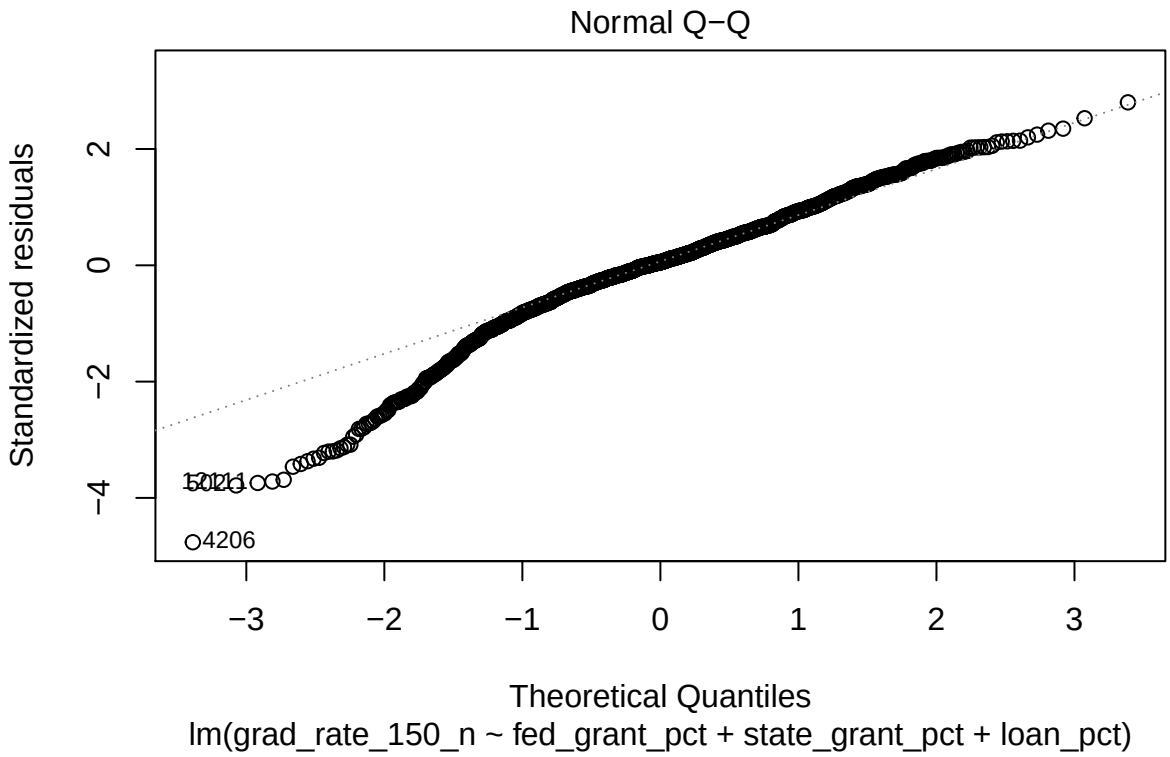
```
# evidence of heteroskedasticity

plot(grad_num150_p_m3, which=1)
```



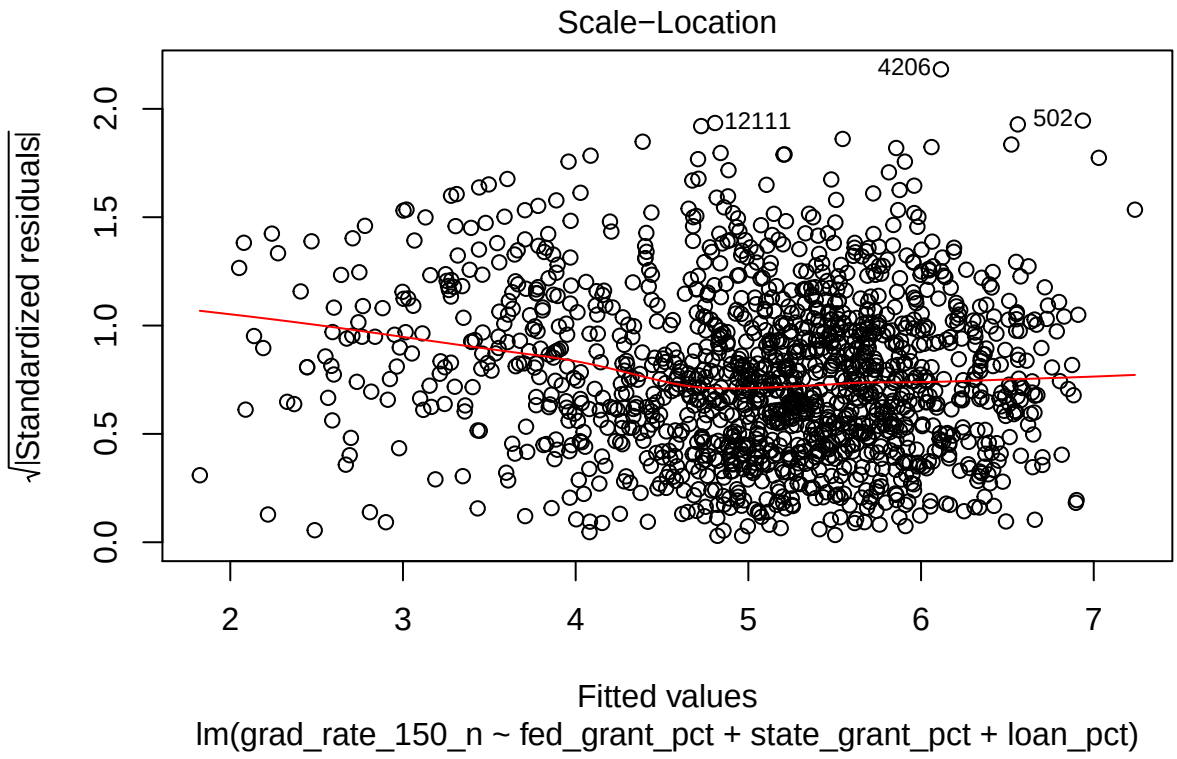

```
# Red residual line much flatter than avg_amount model
# Evidence that ZCM and linearity assumptions OK

plot(grad_num150_p_m3, which=2)
```




```
# Q-Q plot approximately the same as avg_amount model
# Follows the diagonal closely for about 2/3, then bends down at bottom 1/3

plot(grad_num150_p_m3, which=3)
```

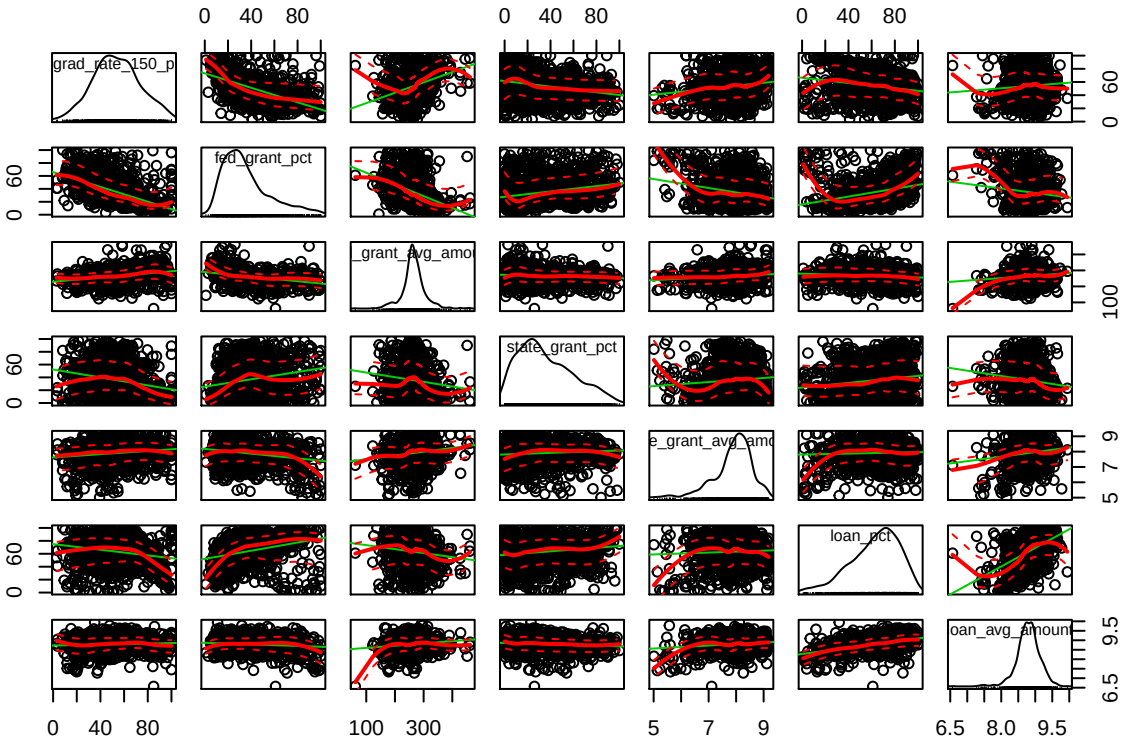



```
# provides evidence of heteroskedasticity
```

We conclude that `grad_num150_p_m3` is the best model for predicting `grad_rate_150_n`. It has better diagnostics than the `grad_num150_a_m3` model. While it shows statistical significance for all 3 coefficients, all of the coefficients are of little practical significance.

3.6 Using grad_rate_150_p as academic success measure

```
scatterplotMatrix(~ grad_rate_150_p + fed_grant_pct + fed_grant_avg_amount +
                    state_grant_pct + state_grant_avg_amount +
                    loan_pct + loan_avg_amount, data = data_TF)
```



From above scatter plot, there are no variables about financial aid which are significant depending on $grad_rate_150_p$. But, we still try to build a linear model among these financial aid related variables to see how insignificant of them.

We proceed with this modeling with some caution. Since the dependent variable is a proportion, a logit transformation may be needed [7]. However, since completion rates tend to cluster in the middle, and the 0 - 1 isn't censored, we think a standard LM is sufficient. Since most of distribution is clustered in the middle, we believe a regular linear model is justified [8].

3.6.1 avg_amount independent variables

Start with most parsimonious model:

```
grad_rate_a_m1 <- lm(grad_rate_150_p ~ fed_grant_avg_amount, data = data_TF)
summary(grad_rate_a_m1)
```

```
##
## Call:
## lm(formula = grad_rate_150_p ~ fed_grant_avg_amount, data = data_TF)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -52.315 -12.706   0.154  12.203  63.352
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)          12.29191    3.31051   3.713 0.000213 ***
## fed_grant_avg_amount  0.15706    0.01237  12.695  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.29 on 1415 degrees of freedom
## Multiple R-squared:  0.1022, Adjusted R-squared:  0.1016
## F-statistic: 161.2 on 1 and 1415 DF,  p-value: < 2.2e-16
```

```
bptest(grad_rate_a_m1)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  grad_rate_a_m1
## BP = 25.366, df = 1, p-value = 4.742e-07
```

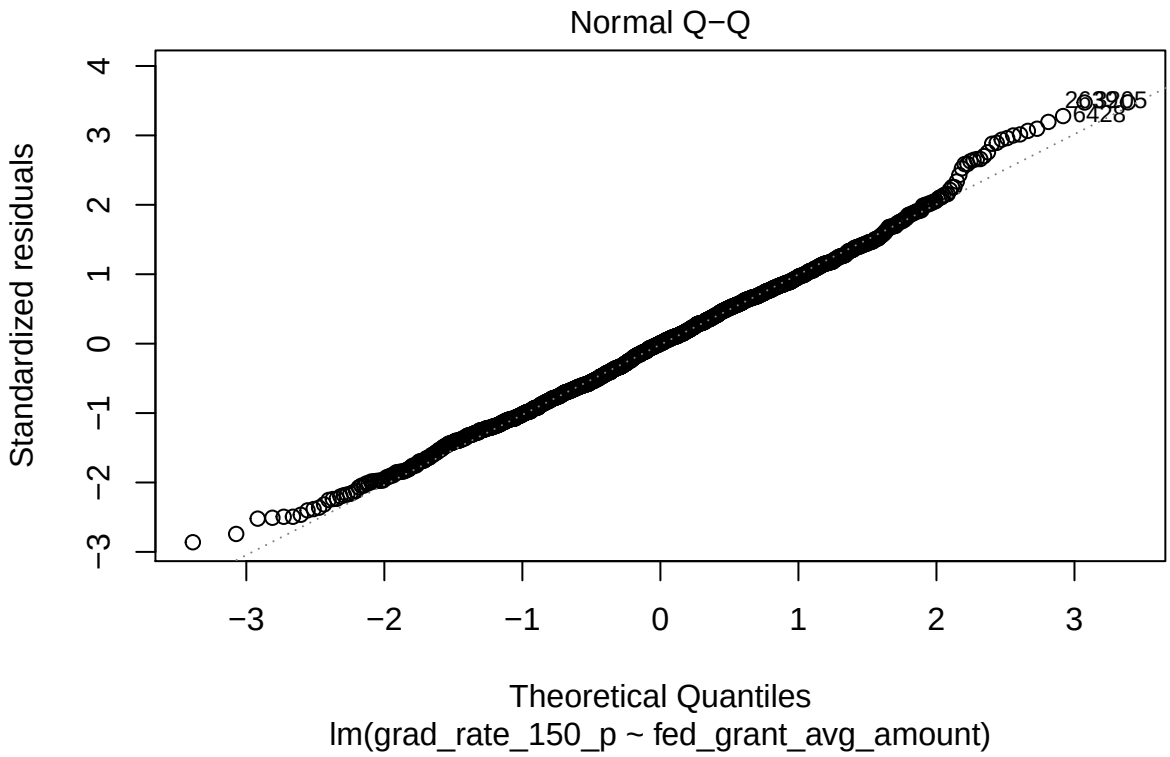
```
# evidence of heteroskedasticity, need robust errors
coeftest(grad_rate_a_m1, vcov=vcovHC)
```

```
##
## t test of coefficients:
##
##                      Estimate Std. Error t value  Pr(>|t|)
## (Intercept)          12.29191    4.19692  2.9288  0.003457 **
## fed_grant_avg_amount  0.15707    0.01546 10.1594 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Significant model. Positive coefficient but low magnitude (no practical significance)
```

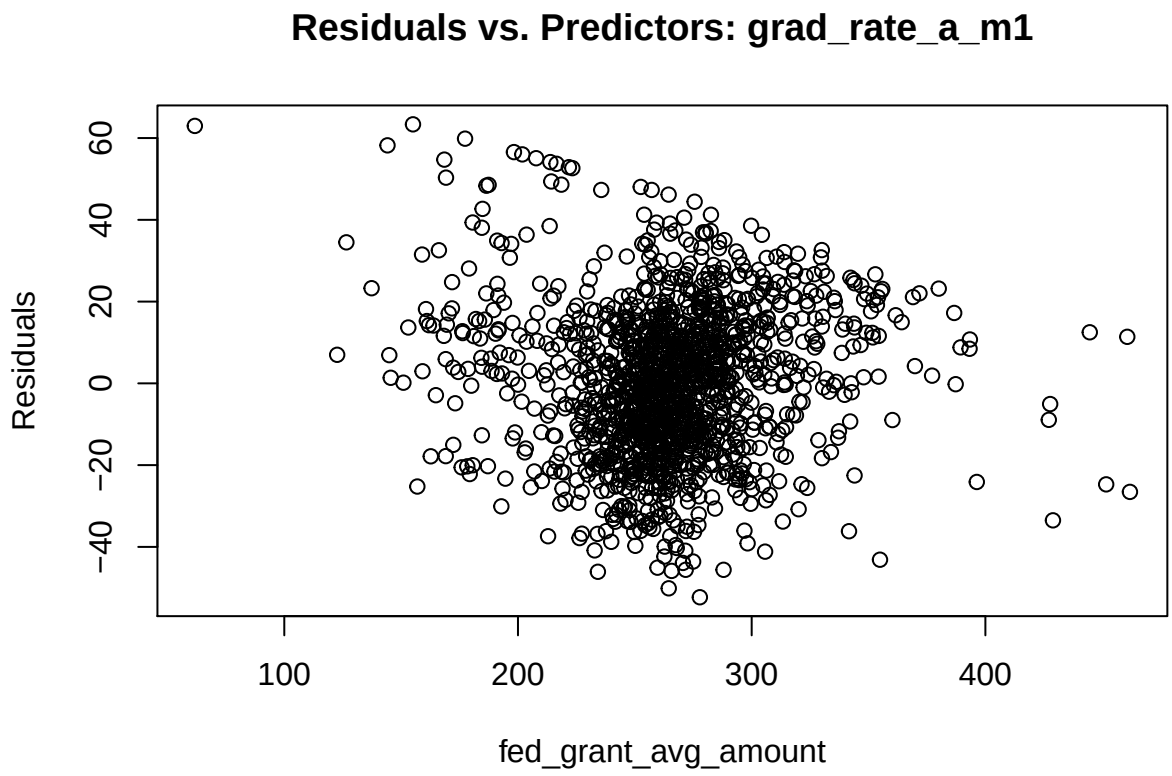
Assess the model diagnostics:

```
plot(grad_rate_a_m1, which=2)
```



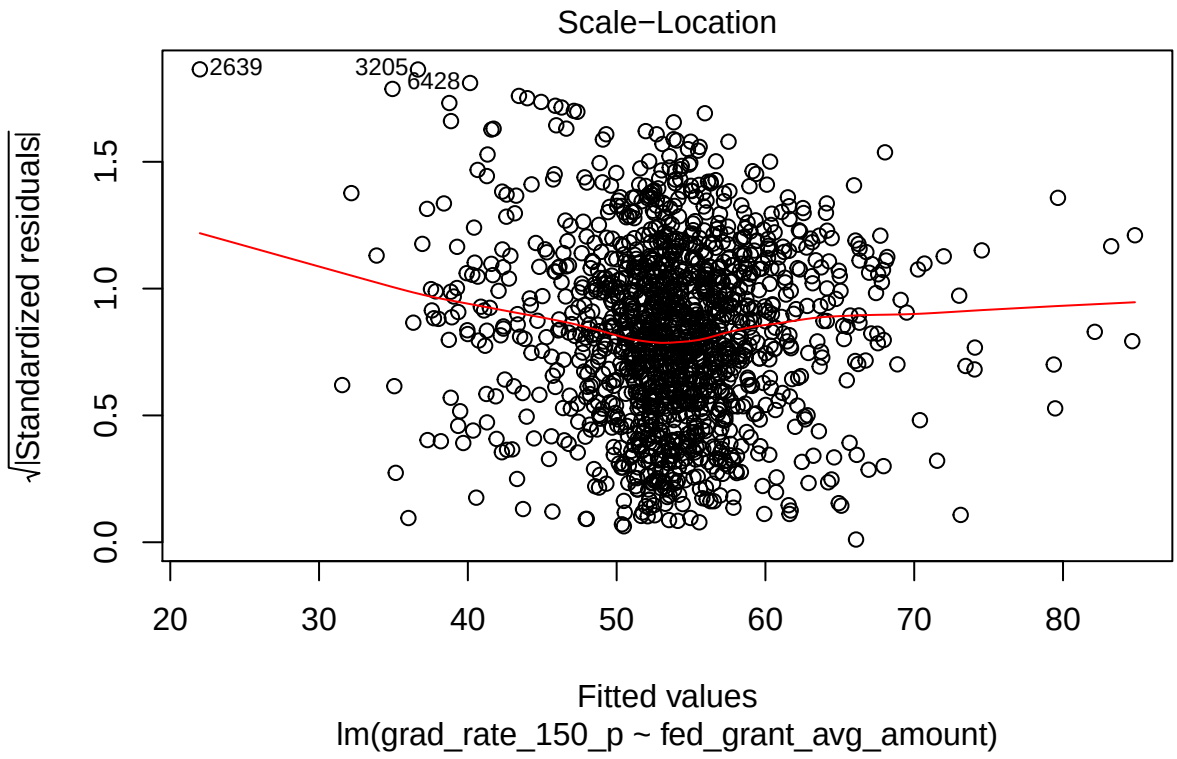

```
# Q-Q plot fairly close to diagonal
# Suggests normality of erros

# Residuals vs. Predictors plot
plot(data_TF$fed_grant_avg_amount, resid(grad_rate_a_m1),
     xlab="fed_grant_avg_amount", ylab="Residuals",
     main="Residuals vs. Predictors: grad_rate_a_m1")
```



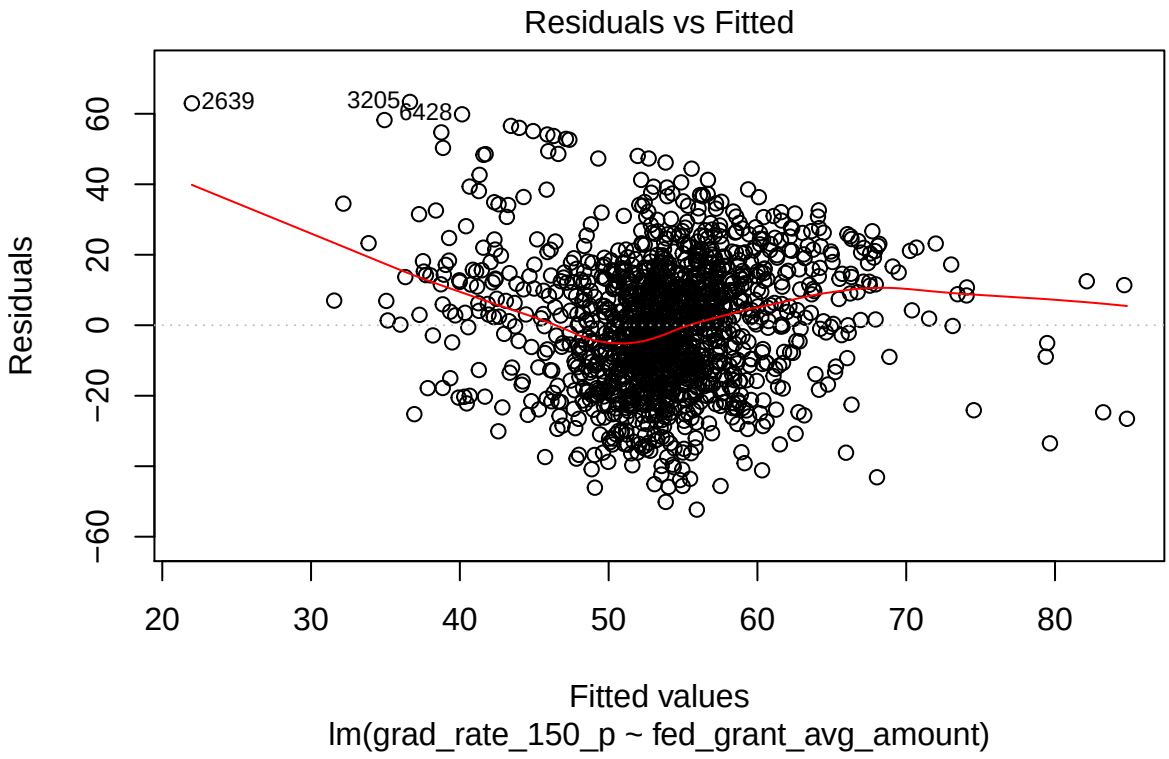

```
# clustering around middle, but roughly even distribution in sign

plot(grad_rate_a_m1, which=3)
```




```
# Scale-Location plot provides evidence of heteroskedasticity,
# since the red line bends up on left half.
# This is consistent with the results of bptest

plot(grad_rate_a_m1, which=1)
```




```
# Some non-flatness in red residual line and overall downward trend
```

Build up model by adding variables:

```
grad_rate_a_m2 <- lm(grad_rate_150_p ~ fed_grant_avg_amount
                     + state_grant_avg_amount, data = data_TF)
coeftest(grad_rate_a_m2, vcov=vcovHC)
```

```
##
## t test of coefficients:
##
##                          Estimate Std. Error t value  Pr(>|t|)
## (Intercept)            -11.195940   7.196953 -1.5556      0.12
## fed_grant_avg_amount     0.149141   0.015466  9.6434 < 2.2e-16 ***
## state_grant_avg_amount   3.235569   0.764480  4.2324 2.461e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# statistical significance, somewhat practical as well

bptest(grad_rate_a_m2)
```

```
##
##  studentized Breusch-Pagan test
```

```
##
## data:  grad_rate_a_m2
## BP = 30.332, df = 2, p-value = 2.592e-07
```

```
# evidence of heteroskedasticity

waldtest(grad_rate_a_m2, grad_rate_a_m1, vcov = vcovHC)
```

```
## Wald test
##
## Model 1: grad_rate_150_p ~ fed_grant_avg_amount + state_grant_avg_amount
## Model 2: grad_rate_150_p ~ fed_grant_avg_amount
##   Res.Df Df      F    Pr(>F)
## 1   1414
## 2   1415 -1 17.913 2.461e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We get a statistically significant F-statistic, so we reject the null hypothesis that `state_grant_avg_amount` has no effect. So `grad_rate_a_m2` is preferred over `grad_rate_a_m1`.

Test whether to include `loan_avg_amount`:

```
grad_rate_a_m3 <- lm(grad_rate_150_p ~ fed_grant_avg_amount
                     + state_grant_avg_amount + loan_avg_amount,
                     data = data_TF)
coeftest(grad_rate_a_m3, vcov=vcovHC)
```

```
##
## t test of coefficients:
##
##                           Estimate Std. Error t value  Pr(>|t|)
## (Intercept)            -18.740286  17.886306 -1.0477    0.2949
## fed_grant_avg_amount     0.148183   0.015407  9.6181 < 2.2e-16 ***
## state_grant_avg_amount   3.170018   0.761635  4.1621 3.343e-05 ***
## loan_avg_amount          0.945784   1.883151  0.5022    0.6156
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

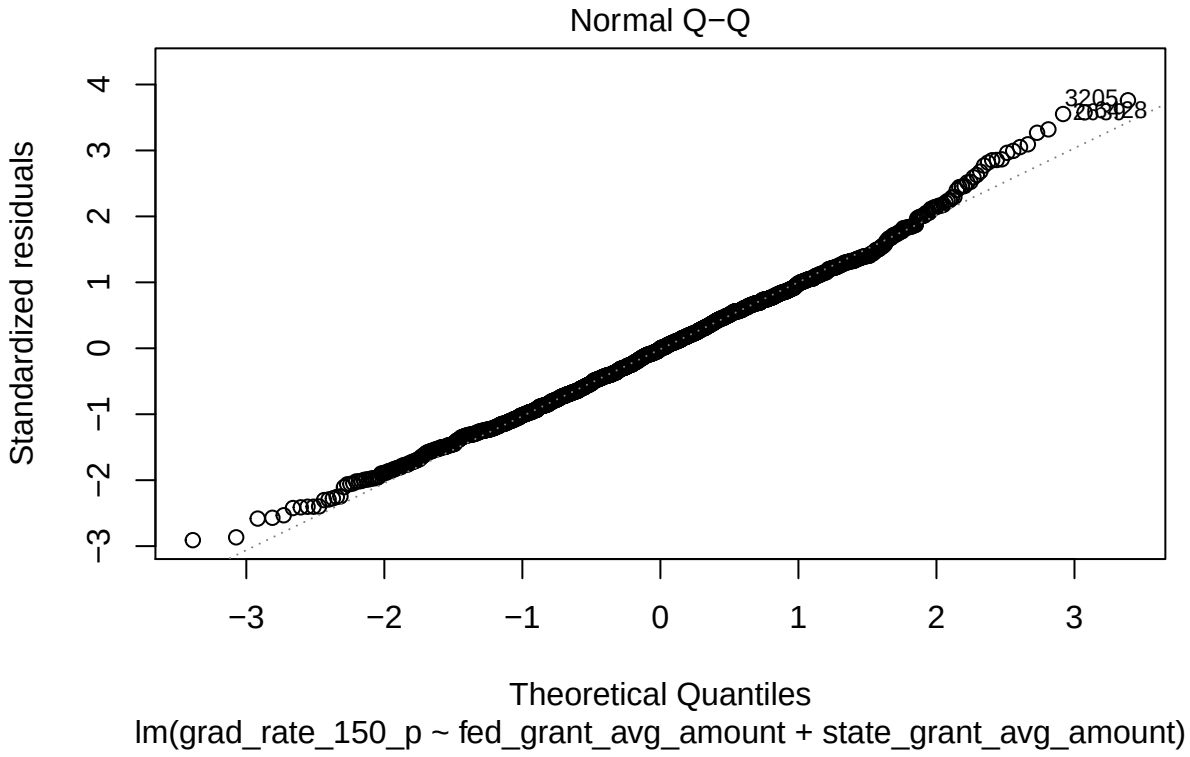
```
# no statistical significance for loan_avg_amount!
waldtest(grad_rate_a_m3, grad_rate_a_m2, vcov = vcovHC)
```

```
## Wald test
##
## Model 1: grad_rate_150_p ~ fed_grant_avg_amount + state_grant_avg_amount +
##     loan_avg_amount
## Model 2: grad_rate_150_p ~ fed_grant_avg_amount + state_grant_avg_amount
##   Res.Df Df      F Pr(>F)
## 1   1413
## 2   1414 -1 0.2522 0.6156
```

We get a statistically insignificant F-statistic. So we cannot reject the null hypothesis that `loan_avg_amount` has no effect. So `grad_rate_a_m2` is preferred.

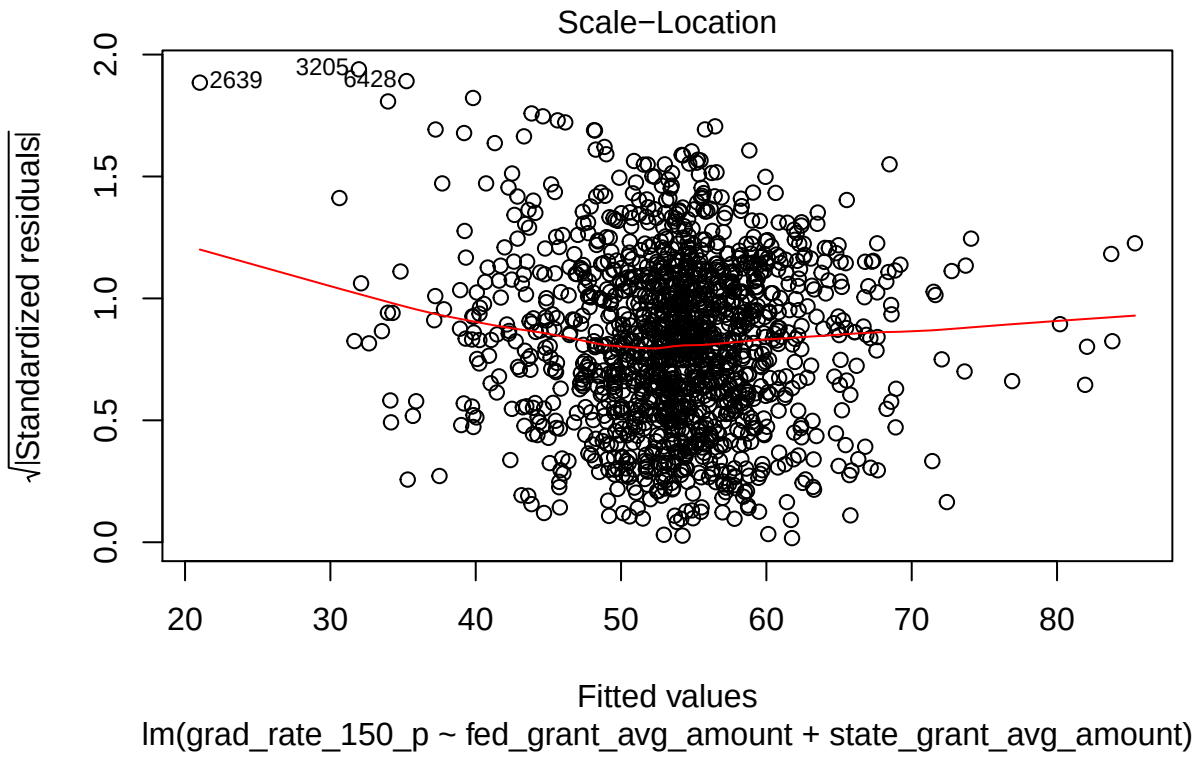
Model diagnostics for grad_rate_a_m2:

```
plot(grad_rate_a_m2, which=2)
```



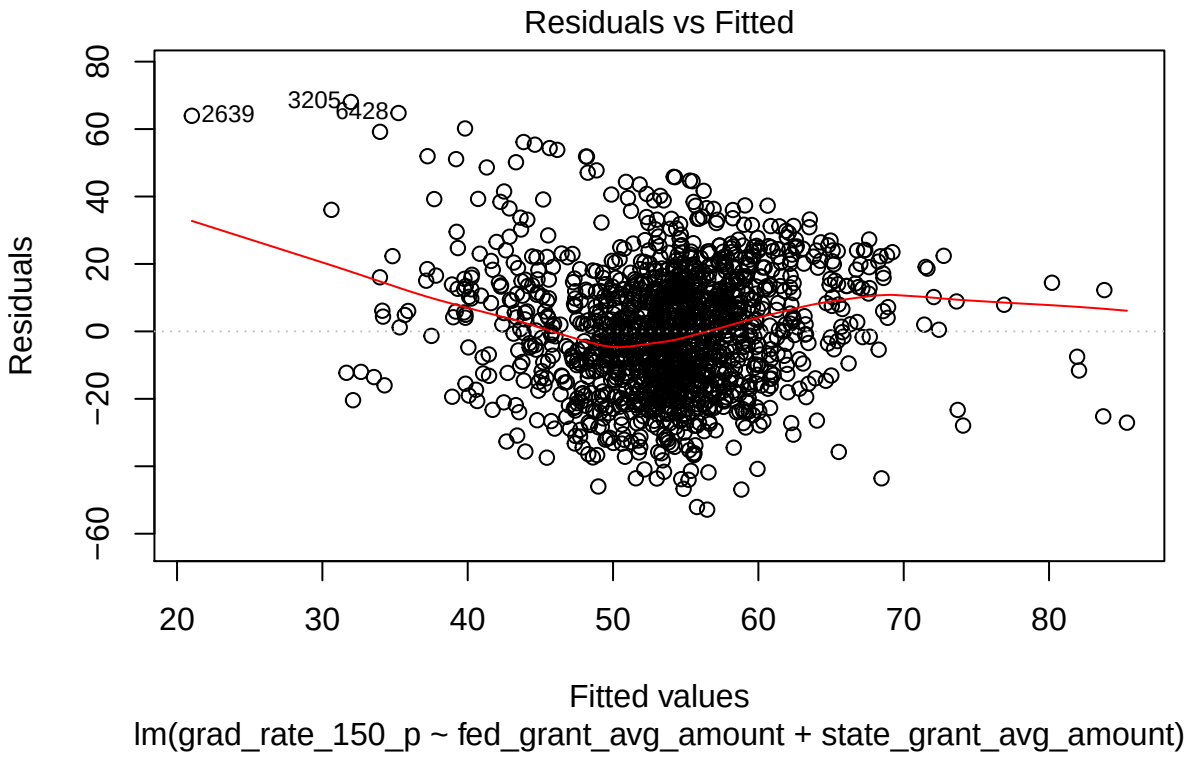

```
# Q-Q plot fairly close to diagonal
# Suggests normality of erros

plot(grad_rate_a_m2, which=3)
```




```
# Scale-Location plot provides evidence of heteroskedasticity,
# since the red line bends up on left half.
# This is consistent with the results of bptest

plot(grad_rate_a_m2, which=1)
```




```
# Residuals vs Fitted plot has sharp bend upward on left side
# Residuals are mostly positive, except in the middle of the fitted values
# May not follow zero conditional mean assumption
```

So despite the promising numbers from the F-statistic `grad_rate_a_m2`, we do not use this as evidence to reject our overall null hypothesis, due to poor diagnostics and strong suspicion of ZCM violation.

3.6.2 Percentage independent variables

Start with most parsimonious model:

```
grad_rate_p_m1 <- lm(grad_rate_150_p ~ fed_grant_pct, data = data_TF)
summary(grad_rate_p_m1)
```

```
##
## Call:
## lm(formula = grad_rate_150_p ~ fed_grant_pct, data = data_TF)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -52.894 -10.267  -1.159   9.125  79.471
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)   73.05350    0.86654   84.31   <2e-16 ***
## fed_grant_pct -0.54713    0.02153  -25.41   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16 on 1415 degrees of freedom
## Multiple R-squared:  0.3134, Adjusted R-squared:  0.3129
## F-statistic: 645.9 on 1 and 1415 DF,  p-value: < 2.2e-16
```

```
bptest(grad_rate_p_m1)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  grad_rate_p_m1
## BP = 36.377, df = 1, p-value = 1.626e-09
```

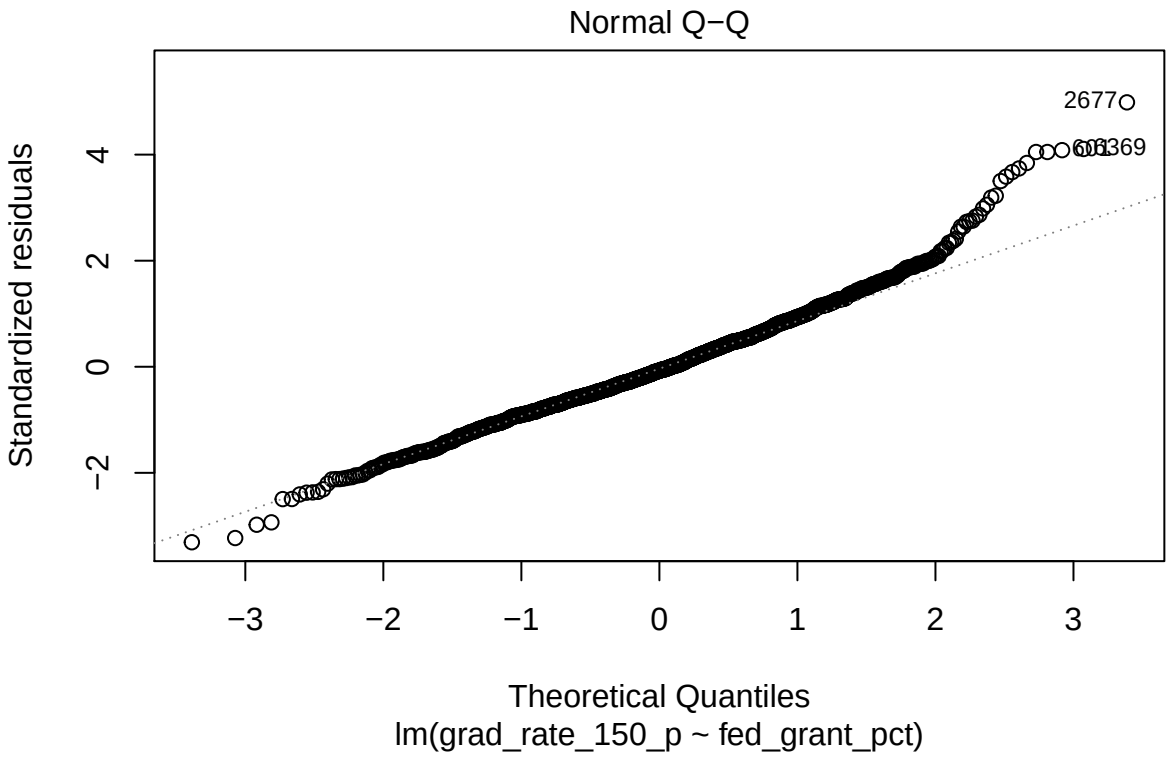
```
# evidence of heteroskedasticity
coeftest(grad_rate_p_m1, vcov=vcovHC)
```

```
##
## t test of coefficients:
##
##                Estimate Std. Error t value  Pr(>|t|)
## (Intercept)   73.053501   0.940546  77.671 < 2.2e-16 ***
## fed_grant_pct -0.547127   0.026998 -20.265 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# stat significant, negative coefficient, but low magnitude
```

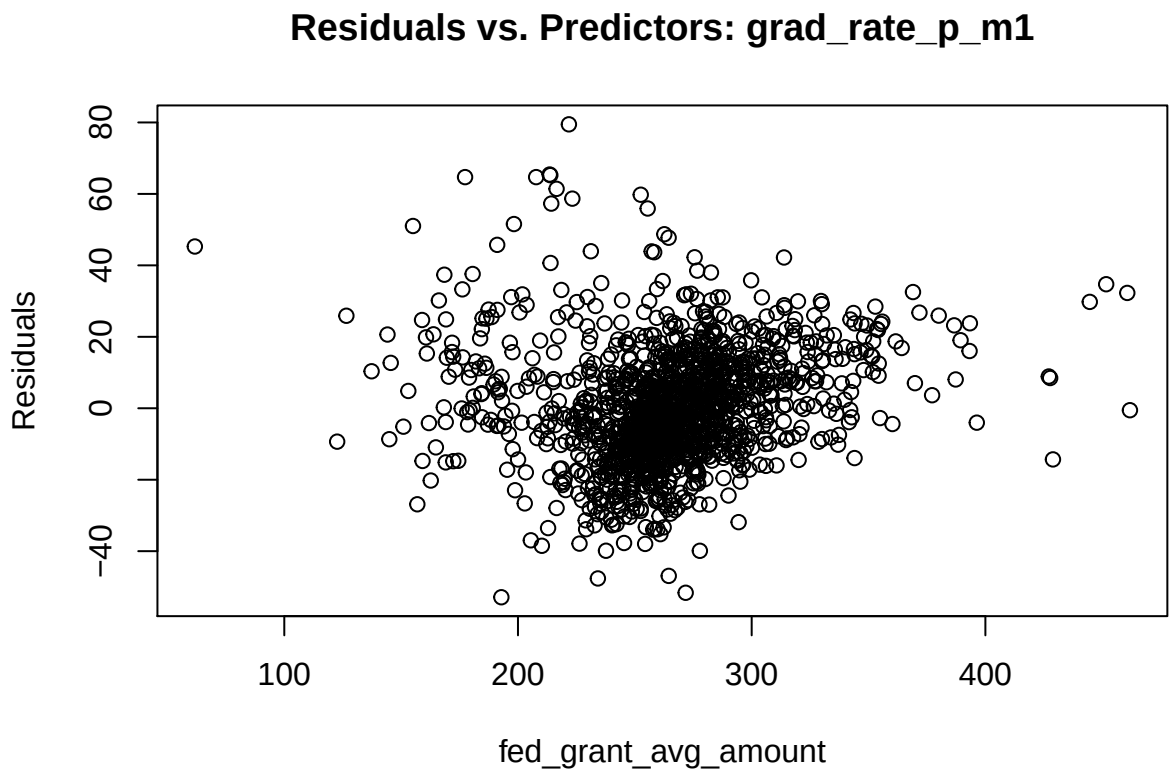
Assess model diagnostics:

```
plot(grad_rate_p_m1, which=2)
```



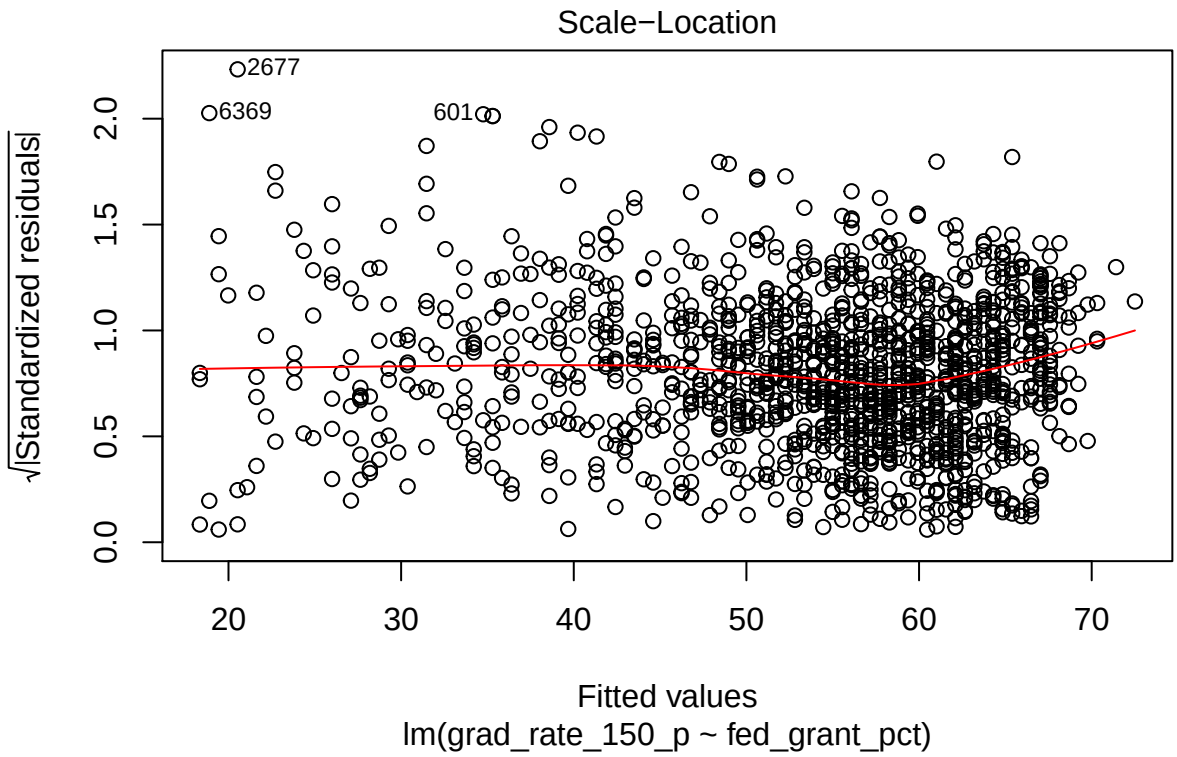

```
# Q-Q plot fairly close to diagonal, only curls away at top
# Suggests normality of erros

# Residuals vs. Predictors plot
plot(data_TF$fed_grant_avg_amount, resid(grad_rate_p_m1),
     xlab="fed_grant_avg_amount", ylab="Residuals",
     main="Residuals vs. Predictors: grad_rate_p_m1")
```



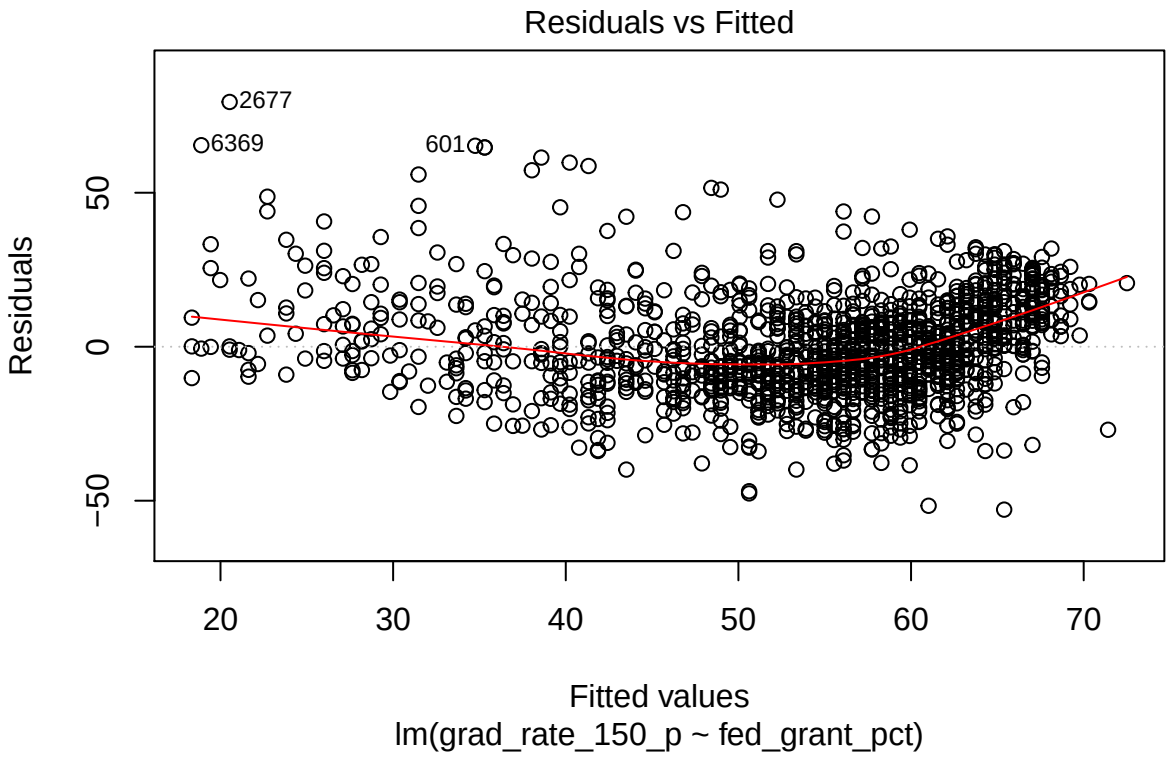

```
# clustering around middle, but roughly even distribution in sign

plot(grad_rate_p_m1, which=3)
```




```
# Scale-Location plot provides evidence of heteroskedasticity,
# Red line relatively flat

plot(grad_rate_p_m1, which=1)
```




```
# Residual line fairly flat and close to 0. Slight u-shape
```

Build up model. Assess whether `state_grant_pct` should be included:

```
grad_rate_p_m2 <- lm(grad_rate_150_p ~ fed_grant_pct + state_grant_pct,
                     data = data_TF)
bptest(grad_rate_p_m2)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  grad_rate_p_m2
## BP = 79.804, df = 2, p-value < 2.2e-16
```

```
# evidence of heteroskedasticity
coeftest(grad_rate_p_m2, vcov=vcovHC)
```

```
##
## t test of coefficients:
##
##                  Estimate Std. Error  t value  Pr(>|t|)
## (Intercept)     75.763203   1.065642  71.0963 < 2.2e-16 ***
## fed_grant_pct   -0.516710   0.027640 -18.6946 < 2.2e-16 ***
## state_grant_pct -0.106015   0.020308  -5.2204 2.052e-07 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# significant model, both negative signs, low magnitude

waldtest(grad_rate_p_m2, grad_rate_p_m1, vcov = vcovHC)
```

```
## Wald test
##
## Model 1: grad_rate_150_p ~ fed_grant_pct + state_grant_pct
## Model 2: grad_rate_150_p ~ fed_grant_pct
##   Res.Df Df      F    Pr(>F)
## 1   1414
## 2   1415 -1 27.252 2.052e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We get a statistically significant F-statistic, so we reject the null hypothesis that `state_grant_pct` has no effect. So `grad_rate_p_m2` is preferred over `grad_rate_p_m1`.

Test whether `loan_pct` should be included:

```
grad_rate_p_m3 <- lm(grad_rate_150_p ~ fed_grant_pct + state_grant_pct
                     + loan_pct, data = data_TF)
bptest(grad_rate_p_m3)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  grad_rate_p_m3
## BP = 97.941, df = 3, p-value < 2.2e-16
```

```
# evidence of heteroskedasticity
coeftest(grad_rate_p_m3, vcov=vcovHC)
```

```
##
## t test of coefficients:
##
##                  Estimate Std. Error  t value  Pr(>|t|)
## (Intercept)     77.060950   1.745887  44.1386 < 2.2e-16 ***
## fed_grant_pct   -0.509087   0.027933 -18.2255 < 2.2e-16 ***
## state_grant_pct -0.104043   0.020480  -5.0803 4.271e-07 ***
## loan_pct        -0.025867   0.024781  -1.0438    0.2967
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# loan_pct is not statistically significant

waldtest(grad_rate_p_m3, grad_rate_p_m2, vcov = vcovHC)
```

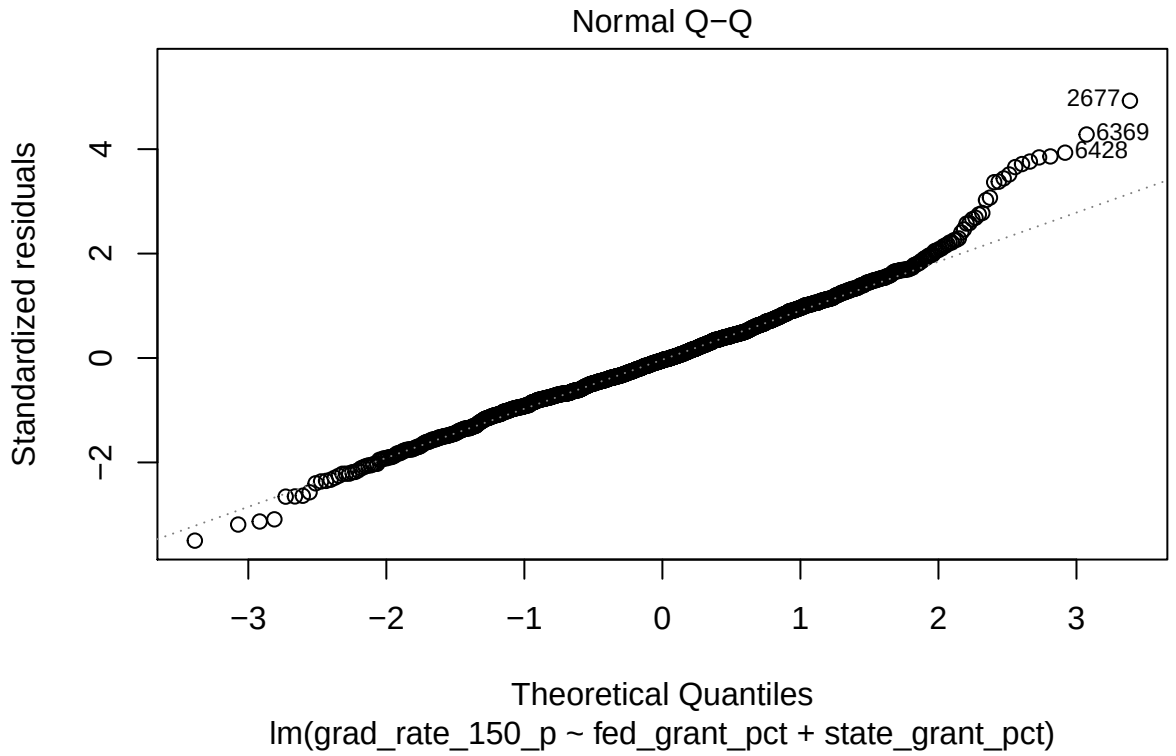
```
## Wald test
```

```
##
## Model 1: grad_rate_150_p ~ fed_grant_pct + state_grant_pct + loan_pct
## Model 2: grad_rate_150_p ~ fed_grant_pct + state_grant_pct
##   Res.Df Df      F Pr(>F)
## 1   1413
## 2   1414 -1 1.0896 0.2967
```

We get a statistically insignificant F-statistic. So we cannot reject the null hypothesis that `loan_pct` has no effect. So `grad_rate_p_m2` is preferred over `grad_rate_p_m3`.

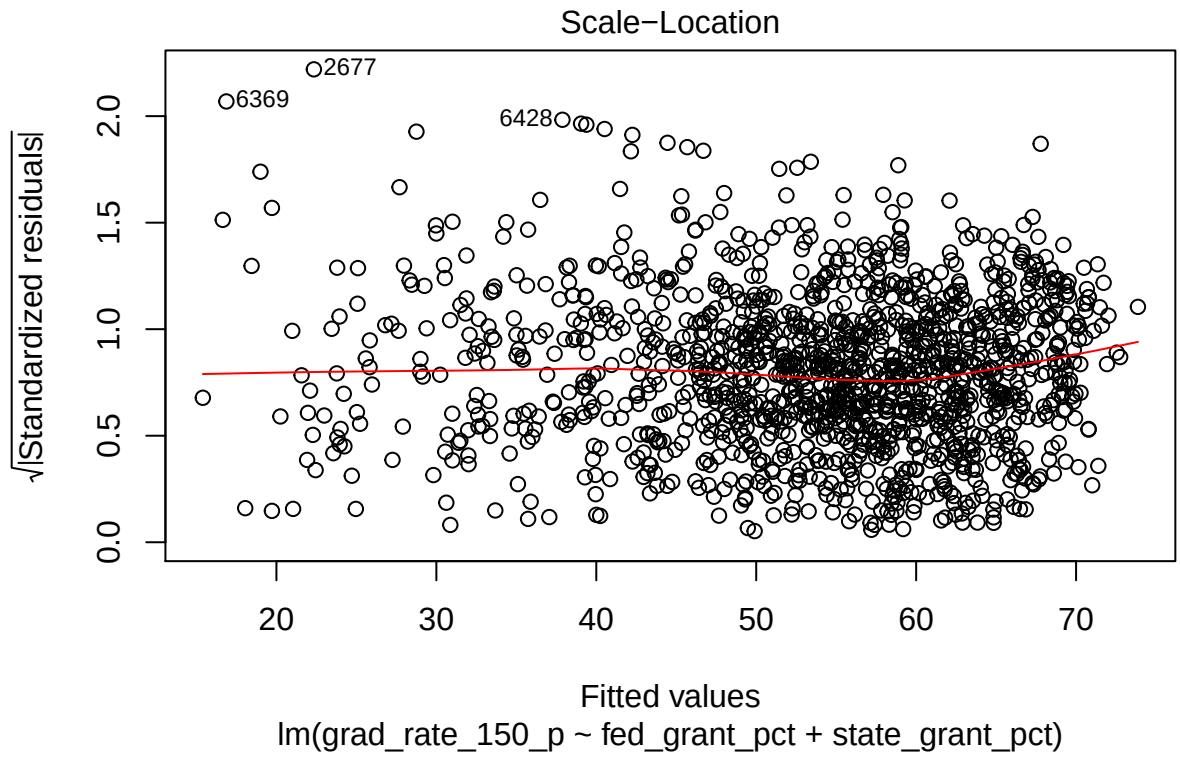
Model diagnostics:

```
plot(grad_rate_p_m2, which=2)
```



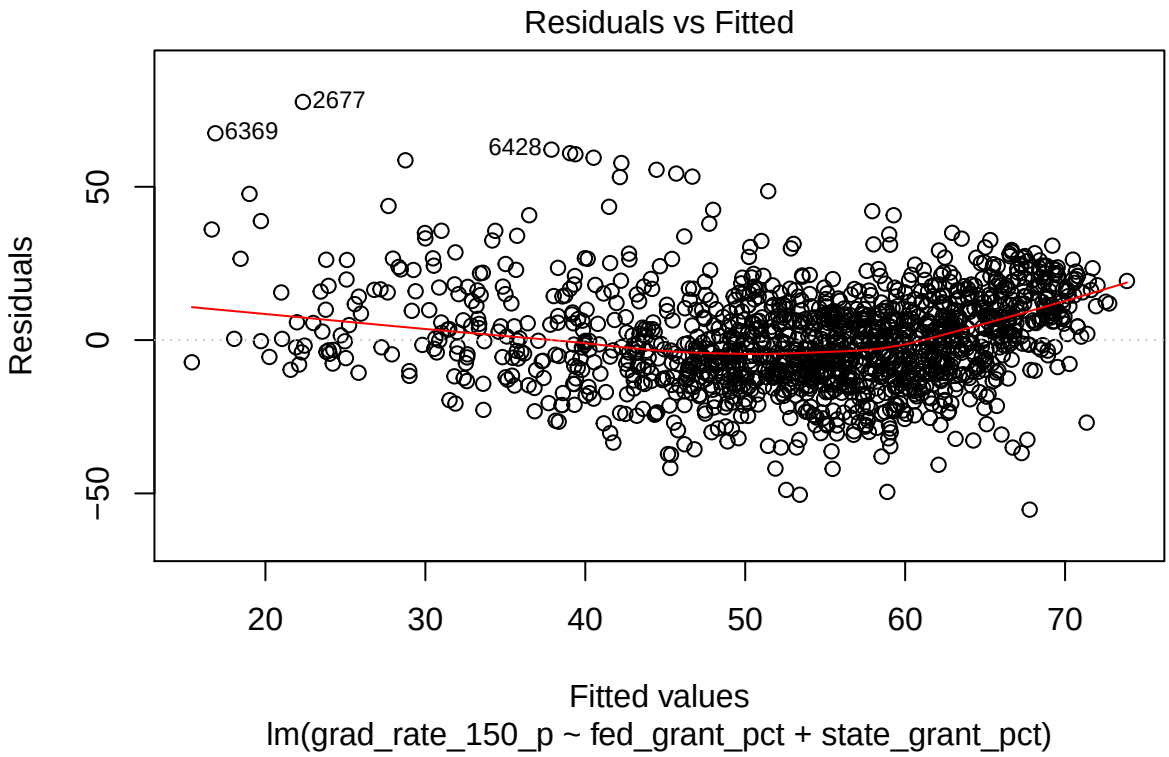

```
# Q-Q plot fairly close to diagonal, except at top
# Suggests normality of erros

plot(grad_rate_p_m2, which=3)
```




```
# Scale-Location plot provides no evidence of heteroskedasticity

plot(grad_rate_p_m2, which=1)
```




```
# Residuals vs Fitted plot fairly flat and close to zero
# ZCM and linearity assumptions OK
```

Due to better diagnostics, we conclude that `grad_rate_p_m2` is a better choice than `grad_rate_a_m2` for predicting `grad_rate_150_p`.

3.7 Part 2 Conclusion:

The percentage independent variables were overall better predictors for both the total number of students that will graduate from an institution and the percentage rate of students graduating from the institution.

In the case of predicting the number of students that will graduate, we found a statistically significant model that that suggested that federal grant percentage and loan percentage have negative effects, state grant percentage has a positive effect on graduation rate. However, these effects are of very little practical significance.

In the case of predicting the percentage rate, we found a statistically significant model that suggested that federal and state grant percentage rate actually have a negative effect on graduation rate. However, this effect is of very little practical significance.

We conclude that there is a negligibly small negative effect on graduation metrics by loans and grants. The negative effect on student success for students who obtained aid , while counter-intuitive, may be accounted for by the large amount of institutional grant in corresponding institutions. It would be interesting to develop a new financial aid measure that incorporate the correlation between institutional grant and tuition, and reexamine the statistical effects of predictor variables studied in this study.

4 References

[1]"What are Title IV Programs?", Federalstudentaid.ed.gov, 2016. [Online]. Available: http://federalstudentaid.ed.gov/site/front2back/programs/programs/fb_03_01_0030.htm. [Accessed: 21- Nov- 2016].

[2]"College Scorecard - College Scorecard - Data.gov", Catalog.data.gov, 2016. [Online]. Available: https://catalog.data.gov/dataset/college-scorecard/resource/70e3a585-0ba2-4ee3-badf-73c015d7041f. [Accessed: 21- Nov- 2016].

[3]"The Condition of Education - Spotlights - 2015 Spotlights - Postsecondary Attainment: Differences by Socioeconomic Status - Indicator May (2015)", Nces.ed.gov, 2016. [Online]. Available: http://nces.ed.gov/programs/coe/indicator_tva.asp. [Accessed: 21- Nov- 2016].

[4]M. Cahalan and L. Perna, "Indicators of Higher Education Equity in the United States", The Pell Institute for the Study of Opportunity in Higher Education and The University of Pennsylvania Alliance for Higher Education and Democracy, 2015. [Online]. Available: http://www.pellinstitute.org/downloads/publications-Indicators_of_Higher_Education_Equity_in_the_US_45_Year_Trend_Report.pdf. [Accessed: 21- Nov- 2016].

[5]D. Desrochers and J. Sun, "The Integrated Postsecondary Education Data System - Delta Cost Project Database", Nces.ed.gov, 2015. [Online]. Available: http://nces.ed.gov/ipeds/deltacostproject/. [Accessed: 02- Dec- 2016].

[6]J. Friedman, "10 Universities With the Most Undergraduate Students", usnews.com, 2016. [Online]. Available: http://www.usnews.com/education/best-colleges/the-short-list-college/articles/2016-09-22/10-universities-with-the-most-undergraduate-students. [Accessed: 02- Dec- 2016].

[7]A. McDowell and N. Cox, "Stata | FAQ: Logit transformation", Stata.com, 2016. [Online]. Available: http://www.stata.com/support/faqs/statistics/logit-transformation/. [Accessed: 05- Dec- 2016].

[8]K. Grace-Martin, "Proportions as Dependent Variable in Regression–Which Type of Model?", Theanalysisfactor.com, 2014. [Online]. Available: http://www.theanalysisfactor.com/proportions-as-dependent-variable-in-regression-whic
[Accessed: 07- Dec- 2016].

5 References

[1]"What are Title IV Programs?", Federalstudentaid.ed.gov, 2016. [Online]. Available: http://federalstudentaid.ed.gov/site/front2back/programs/programs/fb_03_01_0030.htm. [Accessed: 21- Nov- 2016].

[2]"College Scorecard - College Scorecard - Data.gov", Catalog.data.gov, 2016. [Online]. Available: https://catalog.data.gov/dataset/college-scorecard/resource/70e3a585-0ba2-4ee3-badf-73c015d7041f. [Accessed: 21- Nov- 2016].

[3]"The Condition of Education - Spotlights - 2015 Spotlights - Postsecondary Attainment: Differences by Socioeconomic Status - Indicator May (2015)", Nces.ed.gov, 2016. [Online]. Available: http://nces.ed.gov/programs/coe/indicator_tva.asp. [Accessed: 21- Nov- 2016].

[4]M. Cahalan and L. Perna, "Indicators of Higher Education Equity in the United States", The Pell Institute for the Study of Opportunity in Higher Education and The University of Pennsylvania Alliance for Higher Education and Democracy, 2015. [Online]. Available: http://www.pellinstitute.org/downloads/publications-Indicators_of_Higher_Education_Equity_in_the_US_45_Year_Trend_Report.pdf. [Accessed: 21- Nov- 2016].

[5]D. Desrochers and J. Sun, "The Integrated Postsecondary Education Data System - Delta Cost Project Database", Nces.ed.gov, 2015. [Online]. Available: http://nces.ed.gov/ipeds/deltacostproject/. [Accessed: 02- Dec- 2016].

[6]J. Friedman, “10 Universities With the Most Undergraduate Students”, usnews.com, 2016. [Online]. Available: http://www.usnews.com/education/best-colleges/the-short-list-college/articles/2016-09-22/10-universities-with-the-most-undergraduate-students. [Accessed: 02- Dec- 2016].